Cross-validation and Peeling Strategies for Survival Bump Hunting using Recursive Peeling Methods

Jean-Eudes Dazard * Michael Choe † Michael LeBlanc ‡ J. Sunil Rao §

November 24, 2015

Abstract

We introduce a framework to build a survival/risk bump hunting model with a censored time-to-event response. Our Survival Bump Hunting (SBH) method is based on a recursive peeling procedure that uses a specific survival peeling criterion derived from non/semi-parametric statistics such as the hazards-ratio, the log-rank test or the Nelson–Aalen estimator. To optimize the tuning parameter of the model and validate it, we introduce an objective function based on survival or prediction-error statistics, such as the log-rank test and the concordance error rate. We also describe two alternative cross-validation techniques adapted to the joint task of decision-rule making by recursive peeling and survival estimation. Numerical analyses show the importance of replicated cross-validation and the differences between criteria and techniques in both low and high-dimensional settings. Although several non-parametric survival models exist, none addresses the problem of directly identifying local extrema. We show how SBH efficiently estimates extreme survival/risk subgroups unlike other models. This provides an insight into the behavior of commonly used models and suggests alternatives to be adopted in practice. Finally, our SBH framework was applied to a clinical dataset. In it, we identified subsets of patients characterized by clinical and demographic covariates with a distinct extreme survival outcome, for which tailored medical interventions could be made. An R package `PRIMsrc` is available on CRAN and GitHub.

Keywords: Exploratory Survival/Risk Analysis, Survival/Risk Estimation & Prediction, Non-Parametric Method, Cross-Validation, Bump Hunting, Rule-Induction Method.

*Center for Proteomics and Bioinformatics, Case Western Reserve University. Cleveland, OH 44106, USA. Corresponding author Email (JED): jxd101@case.edu

†Center for Proteomics and Bioinformatics, Case Western Reserve University. Cleveland, OH 44106, USA.

‡Department of Biostatistics, School of Public Health, University of Washington, Seattle, WA 98195, USA; Public Health Sciences, Fred Hutchinson Cancer Research Center, Seattle, WA 98109.

§Division of Biostatistics, Dept. of Epidemiology and Public Health, The University of Miami. Miami, FL 33136, USA.

1 Introduction

Non-Parametric Methods for Bump Hunting

The search for data structures in the form of bumps, modes, components, clusters or classes are important as they often reveal underlying phenomena leading to scientific discoveries. It is a difficult and central problem, applicable to virtually all sort of exact and social sciences with practical applications in various fields such as finance, marketing, physics, astronomy and biology.

It is common to treat the task of finding isolated data structures with the help of a response function as in a regression or classification problem or simply a probabilistic model as in a density estimation problem. Among the non-parametric *unsupervised* methods, this can be done by testing modality [10, 35, 58, 61], using nonparametric mixture models (see e.g. [7] for a review), pattern recognition or clustering. However, beside the limitations or problems encountered by these methods in higher dimensional settings, model fitting e.g. of finite mixture models is challenged by the estimation of the *true* number of components [17]. A similar situation exists for clustering procedures where the *true* number of clusters is unknown. Moreover, unsupervised methods may also fail to capture true data structures simply by ignoring a response if available [17]. Although non-parametric *supervised* approaches such as, for instance, decision trees and their ensemble versions [8, 9], do not have this drawback, these classification and regression procedures may also perform poorly [17] since they are designed to work when the true number of classes is fixed or assumed in advance.

Exploratory supervised bump hunting procedures are among the few non-parametric methods that have been proposed to address this problem. These methods seek bump supports (possibly disjoint) of the input space of multi variables where a target function (e.g. a regression or density function) is on average larger (or lower) than its average over the entire input space. They cover tasks such as: (i) Mode(s) Hunting, (ii) Local/Global Extremum(a) Finding, (iii) Subgroup(s) Identification, (iv) Outlier(s) Detection. One known as the Patient Rule Induction Method (PRIM) was initially introduced by Friedman & Fisher [31] and later formalized by Polonik [59]. Essentially, the method is a recursive peeling algorithm that explores the input space to find rectangular solution regions where the response is expected to be larger on average. Some interesting features common and distinct to decision trees such as Classification and Regression Trees (CART) [9] help describe PRIM. As a rule-induction method like CART, PRIM generates simple decision rules describing the solution region of interest. Further, like CART, PRIM is a non-parametric procedure, algorithmic in nature (backwards fitting recursive algorithm), which makes few statistical assumptions about the data. Although PRIM does not explicitly state a model as CART, one can be formulated [36, 73]. Both algorithms/models have the possibility to recover complex interactions between input variables. Basic difference between the two methods lies in their approach and goal (reviewed in section 2.2.1).

To date, only a few extensions of the original PRIM work have been done: This includes a Bayesian model-assisted formulation of PRIM [73], a boosted version of PRIM based on Adaboost [72], an extension of PRIM to censored responses [47, 48] and to discrete variables [39]. Although PRIM is intrinsically multivariate, it was uncertain from the original work how the algorithm would perform in ultra high-dimension where collinearity [30, 31] and sparsity abound. So, recently, an interesting body of work studied when and why the Principal Component space can be used effectively to optimize the response-predictor relationship in bump hunting. This was first addressed in [17], where the computational details of such an approach were laid out for high-dimensional settings. Further, focusing on the properties of PRIM, authors demonstrated using basic geometrical arguments how the PC rotation of the predictor space alone can generate "improved" bump estimates [19, 20]. These developments have important implications for general supervised learning theory and practical applications. In fact, [17] first used a sparse PC rotation for improving bump hunting in the context of high dimensional genomic predictors and later showed how this approach can be used to find additional heterogeneity in terms of survival outcomes for colon cancer patients ([18]).

Model Development and Validation in Discovery-Based Research

The primary problem encountered in discovery-based research has been non-reproducible results. For instance, early biomarker discovery studies using modern high-throughput datasets with large number of features have often been characterized by false or exaggerated claims and eventually disappointment when original results could not be reproduced in an independent study [21, 23, 28, 33, 49, 54, 63, 64, 66]. Sadly, these results have been published even in high-profile journals and considered to provide definitive conclusions for both clinical care and biology. The problems of model reliability and reproducibility have usually been characterized by issues of severe model over-fitting, biased model parameter estimates and under-estimated errors. This has been attributed to a lack of proper rules to assess the analytical validity of studies simply because they were either under-developed or not routinely/correctly applied [57, 60]. This problem first received the attention of statisticians (see for instance reviews on guidelines and checklists [6, 23, 52]) as well as editors and US regulators lately [53].

Meanwhile, considerable development work has been done in the fields of feature selection, predictive model building and model validation to resolve the aforementioned issues. Recent developments include strategies such as variable/feature selection, dimension reduction, coefficient shrinkage and regularization. The challenge is obviously more acute in the context of high-dimensional data where the number of variables greatly exceeds the number of observations (so-called $p \gg n$ paradigm), since usually only a small number of variables truly enter in the model, while the large majority of them just contribute to noise. This noisy situation is even more complicated by the multicollinearity and spurious correlation between variables as well as the endogeneity between variables and model residual errors (see e.g. [29] for a recent review).

A common situation where model reliability and reproducibility arise is when, for instance, model performance estimates are calculated from the same data that was used for model building, eventually resulting in initially promising results, but often non-reproducible [2, 36, 64]. These so-called "resubstitution estimates" are severely (optimistically) biased. Another problematic situation is when not all the steps of model building (such as pre-selection, creation of the prediction rule and parameter tuning) are internal to the cross-validation procedure, thereby creating a selection bias [2, 36, 71]. In addition, findings might not be reproducible even when proper independent sample and validation procedures are used. Problems may arise simply because cross-validated estimates are well-known to have large variance, a situation that is obviously more prevalent when few independent observations or small sample size n are used [22, 24, 51].

Predictive Survival/Risk Modeling by Rule-Induction Methods

One important application of survival/risk modeling is to identify and segregate samples for predictive diagnostic and/or prognosis purposes. Direct applications include the stratification of patients by diagnostic and/or prognostic groups and/or responsiveness to treatment. Therefore, survival modeling is usually performed to predict/classify patients into risk or responder groups (not to predict exact survival time) from which one usually derives survival/risk functions estimates (e.g. by Kaplan–Meier estimates). However, for the reasons mentioned above, Kaplan–Meier estimates for the risk groups computed on the same set of data used to develop the survival model may be very biased [55, 71].

In the context of a time-to-event outcome, regression survival trees have proven to be useful. Several developments have been made for fitting decision trees to non-informative censored survival times [1, 11, 13, 32, 45, 46, 62]. Although regression survival trees are powerful techniques to understand for instance patient outcome and for forming multiple prognostic groups, often times interest focuses only on estimating *extreme* survival/risk groups. In this respect, survival bump hunting aims not at estimating the survival/risk probability function over the entire variable space, but at searching regions where this probability is larger (or smaller) than its average over the entire space.

Also, one possible drawback of decision trees is that the data splits at an exponential rate as the space undergo partitioning (typically by binary splits) as opposed to a more patient rate in decision boxes (typically by controlled quantile). In this sense, bump hunting by recursive peeling may be a more efficient way of learning from the data. With the exception of the work of LeBlanc *et al.* on Adaptive Risk Group Refinement [48], it has not been studied whether decision boxes, obtained from box-structured recursive peelings, would yield better estimates for constructing prognostic groups than their

tree-structured counterparts.

Although resampling methods are often useful in assessing the prediction accuracy of classifier models, they are not directly applicable for predictive survival modeling applications. Simon et *al.* have reviewed the literature of such applications and identified serious deficiencies in the validation of survival risk models [23, 65, 66]. They noted for instance that in order to utilize the cross-validation approach developed for classification problems, some studies have dichotomized their survival or disease-free survival data .... The problem on how to cross-validate the estimation of survival distributions (e.g. by Kaplan–Meier curves) is not obvious [65]. In addition, beside Subramanian and Simon's initial study on the usefulness of resampling methods for assessing survival prediction models in high-dimensional data [67], no comparative study has been done for rule-induction methods and specifically recursive peeling methods such as our "Patient Recursive Survival Peeling" method (see section 2.2.7).

Contribution and Scope

Our survival/risk bump hunting model is built upon the regular bump hunting framework, which we extended to accommodate a possibly censored time-to-event type of response. To build our survival/risk bump hunting model, we first describe our "Patient Recursive Survival Peeling" (PRSP) method, a non-parametric recursive peeling procedure, derived from a rule-induction method, namely the Patient Rule Induction Method (PRIM), which we have extended to allow for survival/risk response, possibly censored. In the process, we describe what appropriate survival estimator and statistic may be used as a peeling criteria to fit our survival/risk bump hunting model.

One of the critiques made in the original PRIM work was the lack of validation procedure and measures of significance of solution regions. So, our objective was also to develop a validation procedure for the purpose of model tuning by means of an optimization criterion of model parameters tuning and a resampling technique amenable to the joint task of decision rule making by recursive peeling (i.e. decision-box) and survival estimation. Specifically, we describe here two alternative, possibly repeated, K-fold cross-validation techniques adapted to the task, namely the "Replicated Combined CV" (RCCV) and "Replicated Averaged CV" (RACV). Moreover, we show how to use survival end-points/prediction statistics for the specific goal of model peeling length optimization by cross-validation.

Results support the claim that optimal survival bump hunting models may be reached using appropriate combination of criterion and technique under certain situations, for which we provide guidelines. Finally, we show empirical results from a real dataset application and from simulated data in low- and high-dimension, illustrating the efficiency of our cross-validation and peeling strategies and the adequacy of our survival bump hunting framework in comparison to other available non-parametric survival models.

We do not describe nor discus the specific treatment of dimension-reduction or variable selection in high-dimensional settings for the only reason that the focus of this study is on cross-validation and peeling strategies. Even though the issue of model unreliability is known to be more severe when there is a large number of variables to choose from [64], it is known to persist even in low-dimensional setting [68]. So, we posit that the framework described here is relevant and applicable to both low and high-dimensional situations. Nevertheless, the method does include cross-validation procedures to control model size (# covariates) in addition to model complexity (# peeling steps). It has been tested in multiple (> 20) low and high-dimensional situations where $n \leqslant p$ and even $n \ll p$ (see abstract of application article [15] and our example datasets in our R package `PRIMsrc` [16]) and we show empirical analyses in high-dimensional simulated datasets where $n < p$.

2 Survival Bump Hunting for Exploratory Survival Analysis

2.1 Bump Hunting Model

2.1.1 Notation - Goal

The formal setup of bump hunting is as follows [see also 31, 59]. Let us consider a supervised problem with a univariate output (response) random variable, denoted $\mathbf{y} \in \mathbb{R}$. Further, let us consider a p-dimensional

random vector $\mathbf{X} \in \mathbb{R}^p$ of support S, also called input space, in an Euclidean space. Let us denote the p input variables by $\mathbf{X} = [\mathbf{x}_j]_{j=1}^p$, of joint probability density function $p(\mathbf{X})$ and by $f(\mathbf{x}) = E(\mathbf{y}|\mathbf{X} = \mathbf{x})$ the target function to be optimized (e.g. any regression function or e.g. the p.m.f or p.d.f $f_{\mathbf{X}}(\mathbf{x})$).

Briefly, the goal in bump hunting is to find a sub-space or region ($R \subseteq S$) of the input space within which the average value $\bar{f}_R$ of $f(\mathbf{x})$ is expected to be significantly larger (or smaller) than its average value $\bar{f}_S$ over the entire input space S (Figure 1). In addition, one wishes that the corresponding support (mass) of R, say β_R, be not too small, that is, greater than a minimal support threshold, say $0 < \beta_0 < 1$.

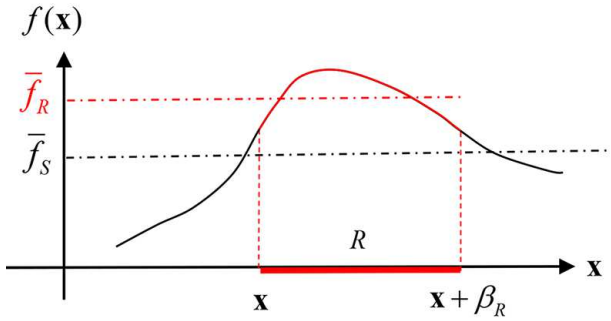


Figure 1: *Schematic representation of bump hunting in the unidimensional case ($p = 1$), where the target function $f(\mathbf{x})$ is a regression function of $\mathbf{x}$ and the estimated region R is a contiguous interval (red segment) corresponding to larger values of $f(\mathbf{x})$ on average. The support β_R of R and the average values $\bar{f}_R$ and $\bar{f}_S$ are shown.*

Formally, in the continuous case of $\mathbf{X}$:

$$\bar{f}_R = \frac{\int_{\mathbf{x}\in R} f(\mathbf{x})p(\mathbf{x})d\mathbf{x}}{\int_{\mathbf{x}\in R} p(\mathbf{x})d\mathbf{x}} \gg \bar{f}_S \tag{1}$$

$$\beta_R = \int_{\mathbf{x}\in R} p(\mathbf{x})d\mathbf{x} \gg \beta_0 \tag{2}$$

In supervised problems with an output variable (response) $\mathbf{y}$, one would seek to characterize the conditional expectation $E(\mathbf{y}|\mathbf{X} = \mathbf{x})$ and infer the properties of the unknown joint probability density function $p(\mathbf{X})$, whereas in the case of unsupervised learning, one would have to directly infer the properties of $p(\mathbf{X})$, e.g. from some density estimate, without the help of a response.

Let S_j be the support of the jth variable $\mathbf{x}_j$, such that the input space can be written as the (Cartesian) outer product space $S = \times_{j=1}^p S_j$. Let $s_j \subseteq S_j$ denotes the unknown subset of values of variable $\mathbf{x}_j$ corresponding to the unknown support of the solution region R. Let $J \subseteq \{1, \ldots, p\}$ be the subset of indices of selected variables in the process. The goal in bump hunting amounts to finding the value-subsets $\{s_j\}_{j\in J}$ of the corresponding variables $\{\mathbf{x}_j\}_{j\in J}$ such that

$$R = \left\{ \bigcap_{j\in J} (\mathbf{x}_j \in s_j) : (\bar{f}_R \gg \bar{f}_S)(\beta_R \gg \beta_0) \right\} \tag{3}$$

2.1.2 Estimates

Since the underlying distribution is not known, the estimates of $\bar{f}_R$ and β_R must be used. Assume a supervised setting, where the outcome response variable is $\mathbf{y} = (y_1 \ldots y_n)^T$ and the explanatory/input variables are $\mathbf{X} = (\mathbf{x}_1 \ldots \mathbf{x}_n)^T$, where each observation is the p-dimensional vector of covariates $\mathbf{x}_i = [x_{i,1} \ldots x_{i,p}]^T$, for $i \in \{1, \ldots, n\}$. Plug-in estimates of the average value $\bar{f}_R$ of the target function $f(\mathbf{x})$ and of the support β_R (eq. 2) of the region R are respectively derived as:

$$\hat{\bar{f}}_R = \frac{1}{n\hat{\beta}_R} \sum_{\mathbf{x}_i\in\hat{R}} y_i = \frac{1}{n\hat{\beta}_R} \sum_{i=1}^n y_i I(\mathbf{x}_i \in \hat{R}) \tag{4}$$

$$\hat{\beta}_R = \frac{1}{n} \sum_{\mathbf{x}_i\in\hat{R}} I(\mathbf{x}_i \in \hat{R}) = \frac{1}{n} \sum_{i=1}^n I(\mathbf{x}_i \in \hat{R}) \tag{5}$$

2.1.3 Remarks

1. The goal amounts to comparing the conditional expectation of the response over the solution region R: $\bar{f}_R = E[f(\mathbf{x})|\mathbf{x} \in R]$ with the unconditional one $\bar{f}_S = E[f(\mathbf{x})]$.
2. Larger target function average $\bar{f}_R$ is associated with smaller support β_R of the region R (Figure 1). So, in practice, there is a trade-off between maximizing $\bar{f}_R$ and maximizing β_R.
3. If the target function to be optimized is for instance the p.m.f or p.d.f $f_{\mathbf{X}}(\mathbf{x})$, then $\Pr(\mathbf{x} \in R)$ is the probability mass/density of a local maximum and the task is equivalent to a mode(s) hunting.
4. In the case of real-valued inputs, the entire input space is the p-dimensional outer product space $S \subseteq \mathbb{R}^p$; the support S_j of each individual input variable (and of each corresponding value-subset s_j) is the usual interval of the form $S_j = \left[t_j^-, t_j^+\right] \subset \mathbb{R}$ for $j \in J$; the solution region R has the shape of a $|J|$-dimensional hyper-rectangle in $\mathbb{R}^{|J|}$, called a *box*, denoted B, which can be written as the outer product of $|J|$ intervals of the form $B = \times_{j \in J}[t_j^-, t_j^+]$.
5. In general, region R could be any smooth shape (e.g. a convex hull) possibly disjoint. Describing or modeling such region would be difficult in high dimension and especially when the number of variables is larger than the number of observations ($p \gg n$ paradigm). In general, there is a trade-off between the goodness of fit and the interpretability of the inferences that we want to make. Here, we focus on interpretable models based on rectangular boxes in the input space of variables. Typically these rectangular boxes are aligned to the coordinate axes, but an immediate extension is to use linear combination rules of variables, i.e. a rotated space of input variables, such a the principal components space. We have showed that this strategy may provide a more favorable space to learn from the data (see for instance [17–20]).

2.1.4 Estimation by the Patient Rule Induction Method (PRIM)

The Patient Rule Induction Method (PRIM) is used to get the region estimate $\hat{R}$ with corresponding support estimate $\hat{\beta}_R$ and conditional output response mean estimate $\hat{\bar{f}}_R$. Essentially, the method is one of recursive peeling/pasting algorithm (a discrete version of the steepest ascent method) that explores the input space solution region, where the response is expected to be larger on average. The method generates a sequence of boxes that collectively cover the region estimate $\hat{R}$. The way the space is covered and the box induction is done as well as how the patience and stopping rules are controlled is detailed in the original article of Friedman & Fisher [31], later formalized by Polonik & Wang [59].

Covering - Coverage Stopping Rule. A sequence of boxes $\{B_m\}_{m=1}^M$ is generated from the data $\{\mathbf{x}_i, y_i\}_{i=1}^n$ to collectively cover the solution region R. Starting from an initial box B_1 that covers all the data, the box sequence construction algorithm is recursively applied to subsets of the data as follows. At the mth iteration ($m > 1$), a box B_m is induced (by the top-down peeling algorithm - see next) using the data remaining after removal of all the observations contained in the previous boxes: $\{(y_i, \mathbf{x}_i) \colon \mathbf{x}_i \notin \bigcup_{r=1}^{m-1} B_r\}$. At the Mth iteration of the covering loop, the box sequence $\{B_m\}_{m=1}^M$ stops either (i) when the estimated individual box support $\hat{\beta}_M$ becomes too small, say less than an arbitrary threshold $0 < \beta_0 < 1$ expressed as a fraction of the entire data: $\hat{\beta}_M < \beta_0$, where, or (ii) when the estimated box output mean $\hat{\bar{y}}_M$ becomes too small, say $\hat{\bar{y}}_M < \bar{y}$ where $\bar{y} = \frac{1}{n}\sum_{i=1}^n y_i$ is the global mean, where

$$\hat{\beta}_M = \frac{1}{n}\sum_{i=1}^{n} I\left(\mathbf{x}_i \in B_M \;\&\; \mathbf{x}_i \notin \bigcup_{m=1}^{M-1} B_m\right)$$

$$\hat{\bar{y}}_M = \frac{1}{n\hat{\beta}_M}\sum_{i=1}^{n} y_i I\left(\mathbf{x}_i \in B_M \;\&\; \mathbf{x}_i \notin \bigcup_{m=1}^{M-1} B_m\right)$$

Box Induction. To induce the box B_m at the mth iteration ($m > 1$), the top-down peeling algorithm generates a subsequence of nested boxes $\{B_{m,l}\}_{l=1}^L$ starting from an initial box $B_{m,1}$ that covers all the data remaining at the mth iteration of the covering loop. How L is estimated is the subject of section 3.2.

At the lth iteration, a sub-box is peeled off (see next) from within the current box $B_{m,l}$ to produce the next smaller box $B_{m,l+1}$. The particular sub-box $b^*_{m,l}$ is chosen to yield the largest box output mean value $\bar{y}_{m,l+1}$ within the next box $B_{m,l+1}$, such that:

$$\begin{aligned}
&\bar{y}_{m,l+1} = \frac{1}{n\hat{\beta}_{m,l+1}} \sum_{i=1}^{n} y_i I\left(\mathbf{x}_i \in B_{m,l+1} \;\&\; \mathbf{x}_i \notin \bigcup_{b=1}^{l} B_{m,b}\right) \\
&B_{m,l+1} = B_{m,l} \backslash b^*_{m,l} \quad \text{, where} \\
&b^*_{m,l} = \text{argmax}_{b_{m,l} \in C(b_{m,l})} \left[\bar{y}_{m,l+1} \colon \mathbf{x}_i \in (B_{m,l} \backslash b_{m,l})\right]
\end{aligned}$$

where $C(b_{m,l})$ represents the class of potential sub-boxes $b_{m,l}$ eligible for removal at step or generation (m, l) and '\' represents the set minus operator. The current box $B_{m,l}$ is then updated: $B_{m,l+1} = B_{m,l+1} \backslash b^*_{m,l}$ and the peeling procedure is looped until some stopping rule is met (see next). Because a top-down peeling is a greedy search algorithm, it may cause overfitting, so a bottom-up pasting is applied to the minimal candidate box to repeatedly expand along any edge until the expansion fails to increase the output response average within the box.

Patience - Induction Stopping Rule. There are two important meta-parameters that control the box induction algorithm: (i) the peeling fraction $0 < \alpha_0 < 1$ that controls the degree of patience, and (ii) the minimal box support threshold $0 < \beta_0 < 1$, expressed as a fraction of the whole data that is used in the stopping criterion (see next). Only a quantile α_0 of the data that is in the box $B_{m,l}$ is peeled off at the lth iteration of the peeling loop as follows. Each eligible sub-box $b_{m,l}$ is defined by a single input variable $\mathbf{x}_j$. For real valued variables, there are two eligible sub-boxes $b^-_{j,m,l} \in C(b_{m,l})$ and $b^+_{j,m,l} \in C(b_{m,l})$, which respectively border the lower and upper boundaries of the box $B_{m,l}$ on the jth input variable $\mathbf{x}_j$:

$$\begin{cases} b^-_{j,m,l} = \{\mathbf{x} \colon \mathbf{x}_j < \mathbf{x}_j^{(\alpha_0)}\} \\ b^+_{j,m,l} = \{\mathbf{x} \colon \mathbf{x}_j > \mathbf{x}_j^{(1-\alpha_0)}\} \end{cases}$$

where $\mathbf{x}_j^{(\alpha_0)}$ and $\mathbf{x}_j^{(1-\alpha_0)}$ are respectively the α_0th and $(1-\alpha_0)$th quantiles of the $\mathbf{x}_j$ values. At the Lth iteration of the peeling loop, the box sequence $\{B_{m,l}\}_{l=1}^{L}$ stops when the estimated individual support $\hat{\beta}_{m,L}$ of the last box $B_{m,L}$ becomes too small, say $\hat{\beta}_{m,L} < \beta_0$, where β_0 is an arbitrary minimal box support threshold:

$$\begin{cases} \hat{\beta}_{m,1} = \frac{1}{n}\sum_{i=1}^{n} I(\mathbf{x}_i \in B_{m,1}) & \text{for} \quad L = 1 \\ \hat{\beta}_{m,L} = \frac{1}{n}\sum_{i=1}^{n} I(\mathbf{x}_i \in B_{m,L} \quad \& \quad \mathbf{x}_i \notin \bigcup_{l=1}^{L-1} B_{m,l}) & \text{for} \quad L > 1 \end{cases}$$

Note that, with our notation, the last box $B_{m,L}$ of the subsequence is also the next box of the outer box sequence $\{B_m\}_{m=1}^{M}$. So, $B_{m,L} = B_{m+1}$, and similarly $\hat{\bar{y}}_{m,L} = \hat{\bar{y}}_{m+1}$ and $\hat{\beta}_{m,L} = \hat{\beta}_{m+1}$.

Decision Rules. It is desirable that the solution region R be described in an interpretable form by logical statements involving the value-subset of each selected input variable. The above algorithm results in simple decision rules of the input space, where each box B_m, $m = 1, \ldots, M$, is described by the outer product of the value-subsets $s_{j,m}$ of each individual input variable $\mathbf{x}_j$, for $j \in J$. The idea is to describe the solution region R by a disjunctive rule of M conjunctive subrules of the form $\mathcal{R} = \bigcup_{m=1}^{M} \mathcal{R}_m$, where $\mathcal{R}_m = \{\mathbf{x} \in B_m\} = \bigcap_{j \in J}(\mathbf{x}_j \in s_{j,m})$. In the case of real-valued input variables, each subrule becomes $\mathcal{R}_m = \bigcap_{j \in J}(\mathbf{x}_j \in [t^-_{j,m}, t^+_{j,m}])$ and the solution region R is fully described by the disjunctive rule:

$$\hat{\mathcal{R}} = \bigcup_{m=1}^{M} \hat{\mathcal{R}}_m = \bigcup_{m=1}^{M} \left\{ \bigcap_{j \in J} (\mathbf{x}_j \in [t^-_{j,m}, t^+_{j,m}]) \right\}$$

2.2 Survival Bump Hunting by Recursive Peeling

Assume a supervised problem, where the function of interest is a univariate survival/risk response variable (possibly censored) in a multivariate setting of real-valued (continuous or discrete) input variables/covariates $\mathbf{X} = [\mathbf{x}_j]_{j=1}^{p}$. The goal is to characterize an extreme-survival-response support in the predictor space and identify the corresponding box-defined group of samples using a recursive peeling method derived from the Patient Rule Induction Method (PRIM).

2.2.1 Survival-Specific Peeling Rule

As mentioned in the introduction, rule-induction methods such as decision tree-based methods have proven to be useful to estimate relative risk in groups in the context of a time-to-event outcome. Several methods have been proposed for fitting trees to non-informative censored survival times [1, 11, 13, 32, 45, 46, 62].

Basic differences between decision-tree and decision-box methods lie in their approach and goal. Instead of recursively partitioning the space using specific partitioning and stopping criteria, one proceeds by recursively peeling the space to produce box-shaped regions designed to approximate the solution region R, using specific peeling and stopping criteria (see details in 2.1.4). In decision-trees, a recursive partitioning method attempts to model the target function over the entire data space by generating partitions in which the response averages will be as different as possible, while in decision-boxes, a recursive peeling method generates a box-shaped region in which the response average will be as extreme as possible. So, in contrast to survival decision-trees models, survival bump hunting is not aimed at estimating the survival/risk probability function over the entire covariate space, but at finding regions where this probability is larger than its average over the entire space. Numerical analysis below (4) show comparisons of relative risk estimates obtained from decision-boxes versus those obtained from decision-trees. Other interesting differences lie in the weaknesses and strengths of the outputs and their applications, which we left for discussion (6).

In this section, we describe the use of several candidate survival-specific peeling criteria and discuss their merits or strengths. Most of these criteria are borrowed from the survival splitting rules used to grow regression survival trees [1, 45, 46, 62, 69] or from their ensemble versions [40]. Here, survival-specific peeling criteria are to be used to decide which covariate will be selected to give the best peel between two boxes from two consecutive generations (parent-child descendance) of the box induction/peeling loop in a recursive peeling algorithm (see next section 2.2.7).

To account for censoring, we simply supervise by proxy for extreme time-to-event outcome, turning the censored outcome y into an uncensored "surrogate" outcome z. Using previous notation (section 2.1.4), a peeling at step (m, l) of the box induction/peeling sequence produces a partition of the survival data from the parent box $B_{m,l-1}$ into two partitions, for a given set of covariates: the child box $B_{m,l}$ and its complement. The focus is on selecting a sub-box $b_{m,l}$ at step (m, l) of the box induction/peeling sequence that is to be peeled off from the parent box $B_{m,l-1}$ along one of its faces (i.e. direction of peeling := axis of dimension j) to induce the next child box $B_{m,l}$ and its complement. This is done by maximizing the "surrogate" outcome rate of increase between two consecutive generations of boxes $B_{m,l-1}$ and $B_{m,l}$ of the box induction/peeling sequence. Denote by $z(m, l)$ the box "surrogate" outcome at step or generation (m, l) of the box induction/peeling sequence (Algorithm 1). The rate of increase in $z(m, l)$ at step or generation (m, l) between two consecutive generations of boxes $B_{m,l-1}$ and $B_{m,l}$ is defined as:

$$r(m,l) = \frac{z(m,l) - z(m,l-1)}{\beta_{m,l-1} - \beta_{m,l}} \tag{6}$$

Finally, the particular sub-box $b^*_{m,l}$ that is chosen to yield the largest box increase rate $r(m, l)$ between box $B_{m,l-1}$ and the next one $B_{m,l}$ is such that

$$\begin{aligned} B_{m,l} &= B_{m,l-1} \backslash b^*_{m,l} \quad \text{, where} \\ b^*_{m,l} &= \underset{b_{m,l} \in C(b_{m,l})}{\operatorname{argmax}} \left[r(m,l) \right], \end{aligned} \tag{7}$$

where $C(b_{m,l})$ represents the class of potential sub-boxes $b_{m,l}$ eligible for removal at step or generation (m, l).

2.2.2 Survival Notation and Definitions

Let's denote the two child boxes described above by $\{B_{g,m,l}\}_{g=1}^{2}$, where, by convention, let's decide that subscript $g = 1$ stands for the "in-box" $B_{m,l}$ and $g = 2$ for its complement or "out-of-box". Dropping further step subscripts (m, l) for simplicity, assume that there are n individuals in parent box $B_{m,l-1}$ and n_g in a given child box $B_{g,m,l}$ for fixed $g \in \{1, 2\}$ such that $n = \sum_{g=1}^{2} n_g$. Also, we let $\gamma_i(g) = I(\mathbf{x}_i \in B_{g,m,l})$ be the indicator function of individual subject i within a given child box $B_{g,m,l}$ at step (m, l) for fixed $g \in \{1, 2\}$.

The response variable being subject to censoring, we use the general random censoring model. We focus on a univariate right-censored survival outcome under the assumptions of independent observations, non-competitive risks and random (type-I or -II) non-informative censoring. Denote the *true* survival time (or lifetime/failure time) by the random variable T and the *observed* censoring time by the random variable C, then the *observed* survival time is the random variable $Y = \min(T, C)$. Also, under our assumptions, C is assumed to be independent of T conditionally on covariates $\mathbf{X}$. Let the *observed* event indicator random variable be $\Delta = I(T \leqslant C)$.

For each observation $i \in \{1, \ldots, n\}$ in parent box $B_{m,l-1}$, the true survival time, observed censoring time, observed survival time and observed indicator event are the realizations denoted by T_i, C_i, $Y_i = \min(T_i, C_i)$ and $\delta_i = I(T_i \leqslant C_i)$, respectively. Also, denote by $t_{(1)} < t_{(2)} < \ldots < t_{(N)}$ for $N \leqslant n$ the *distinct ordered* event times of death (not counting censoring times) in parent box $B_{m,l-1}$. Note that intervals between events $t_{(h)}$ for $h \in \{1, \ldots, N\}$ are not necessarily uniform. Finally, the observed data in parent box $B_{m,l-1}$ consists of $(Y_i, \delta_i, \mathbf{x}_i)$, where $\mathbf{x}_i = [x_{i,1} \ldots x_{i,p}]^T$, for $i \in \{1, \ldots, n\}$.

Let $\delta_{i,g} = \gamma_i(g) I(T_i \leqslant C_i)$ be the observed indicator event of time point T_i for each individual $i \in \{1, \ldots, n_g\}$ in a given child box $B_{g,m,l}$ for $g \in \{1, 2\}$. Also, let $d_{h,g}$ and $n_{h,g}$ be respectively the number of events (deaths) and individuals at risk at time $t_{(h)}$ for $h \in \{1, \ldots, N_g\}$ in a given child box $B_{g,m,l}$ for $g \in \{1, 2\}$, such that $N = \sum_{g=1}^{2} N_g$. For simplicity, let's use the same subscript $h \in \{1, \ldots, N_g\}$ and $i \in \{1, \ldots, n_g\}$ from parent and child boxes for indexing events and individuals, respectively. Note that $n_{h,g}$ is the number of individuals in child box $B_{g,m,l}$ who either have not yet had an event (or been right-censored) just until time $t_{(h)}$ or who had an event at time $t_{(h)}$. Formally, if considering all individuals in child box $B_{g,m,l}$ only, $n_{h,g} = \sum_{i=1}^{n_g} I\left(Y_i \geqslant t_{(h)}\right)$ or, if considering all individuals in parent box $B_{g,m,l-1}$, $n_{h,g} = \sum_{i=1}^{n} \gamma_i(g) I\left(Y_i \geqslant t_{(h)}\right)$. Likewise, one can write $d_{h,g} = \sum_{i=1}^{n_g} \delta_{i,g} I\left(Y_i \geqslant t_{(h)}\right)$, or $d_{h,g} = \sum_{i=1}^{n} \delta_i \gamma_i(g) I\left(Y_i \geqslant t_{(h)}\right)$. Also, denote $d_h = \sum_{g=1}^{2} d_{h,g}$ and $n_h = \sum_{g=1}^{2} n_{h,g}$.

Let $S(t) = \Pr(T \geqslant t)$ be the survival probability that an individual from the population of interest will have a lifetime T free of the event until time t. As usual, denote by $\Lambda(t) = -\log(S(t))$ the corresponding cumulative hazard function and by $\lambda(t) = \frac{d\Lambda(t)}{dt}$ the hazard rate function. To come up with decision-box survival-specific peeling criteria (see next section 2.2.3), the following non/semi-parametric estimators can be used with respect to the box-defined subgroups: the Nelson–Aalen estimator, denoted by $\hat{H}_{g,m,l}(t)$, to estimate the cumulative hazard function; and the hazard rate function estimator derived from a Cox Proportional Hazards (CPH) regression model. By definition, these estimators are given as follows for individuals in a given child box $B_{g,m,l}$, for fixed $g \in \{1, 2\}$, at step (m, l):

$$\hat{H}_{g,m,l}(t) = \sum_{h:t_{(h)} \leqslant t} \frac{d_{h,g}}{n_{h,g}}$$

As usual, the hazard rate function may be estimated by regressing the subject-specific hazard rate on the covariates in a CPH regression model, assuming proportional hazards [12]. With the above notation,

$$\begin{aligned} \hat{\lambda}_{i,g,m,l}(t|\mathbf{x}_i) &= \lambda_0(t) \exp\left[\eta_{g,m,l}(\mathbf{x}_i)\right] \\ &= \lambda_0(t) \exp\left[\eta_{g,m,l}\gamma_i(g)\right] \end{aligned}$$

where the regression function $\eta_{g,m,l}(\mathbf{x}_i) = \boldsymbol{\eta}_{g,m,l}^T \mathbf{x}_i = \sum_{j=1}^{p} \eta_{j,g,m,l} x_{i,j}$ with p-dimensional vectors of regression coefficients $\boldsymbol{\eta}_{g,m,l} = [\eta_{1,g,m,l} \ldots \eta_{p,g,m,l}]^T$ and covariate $\mathbf{x}_i = [x_{i,1} \ldots x_{i,p}]^T$ reduce respectively to a scalar $\boldsymbol{\eta}_{g,m,l} = \eta_{1,g,m,l} = \eta_{g,m,l}$ times a simple box indicator variable $\mathbf{x}_i = x_{i,1} = I(\mathbf{x}_i \in B_{g,m,l}) = \gamma_i(g)$.

2.2.3 Non-Parametric Survival Peeling Criteria

The choice of *uncensored* surrogate outcome $z(m,l)$ in equation 6, that is, which estimator to choose as a box peeling criterion at a peeling step (m,l), is central to the PRSP algorithm (see Algorithm 1). Currently, our Survival Bump Hunting implementation in our R package `PRIMsrc` [16] offers three statistics derived from the above non/semi-parametric estimators: (i) the Log-Rank Test statistic, (ii) the Nelson–Aalen Summary statistic and (iii) the CPH-derived Log Hazard Ratio statistic (assuming proportional hazards).

- The (two-sample) log-rank test can be used at a peeling step (m,l) to compare estimates of the hazard functions from each child box-defined subgroups ("in-box" and its "out-of-box" complement). We recently proposed to use it as a survival-specific box peeling criterion [14]. Using the (two-sample) log-rank test statistic is actually a natural candidate for survival decision-box, having been a well-established concept for splitting trees in survival decision-trees [1, 45, 46, 62, 69] and for being robust in non-proportional hazard settings [46]. The approximate log-rank test introduced by LeBlanc and Crowley can be used instead to greatly reduce computations [46]. Formally, one can derive the Log-Rank Test (LRT) statistic, denoted $\hat{\chi}_{LRT}(m,l)$, for the individuals in a given child box $B_{g,m,l}$, for fixed $g \in \{1,2\}$ (e.g. $g=1$), at step (m,l) as follows:

$$\hat{\chi}_{LRT}(m,l) = \frac{\sum_{h=1}^{N}\left(d_{h,1} - n_{h,1}\frac{d_h}{n_h}\right)}{\sqrt{\sum_{h=1}^{N} n_{h,1}\frac{d_h}{n_h}\left(1-\frac{n_{h,1}}{n_h}\right)\left(\frac{n_h-d_h}{n_h-1}\right)}} \tag{8}$$

- If the Nelson–Aalen estimator is used, one can derive an overall summary statistic across all observed time points Y_i for $i \in \{1,\ldots,n_g\}$ of the individuals in a given child box $B_{g,m,l}$, for fixed $g \in \{1,2\}$ (e.g. $g=1$), at step (m,l). This is done by adding the Nelson–Aalen estimators over all these time points to obtain a so-called Cumulative Hazard Summary (CHS), denoted $\hat{\Lambda}_{CHS}(m,l)$:

$$\begin{aligned}
\hat{\Lambda}_{CHS}(m,l) &= \sum_{i=1}^{n_1} \hat{H}_{1,m,l}(Y_i) \\
&= \sum_{i=1}^{n_1}\left(\sum_{h:t_h \leqslant Y_i} \frac{d_{h,1}}{n_{h,1}}\right) \\
&= \sum_{i=1}^{n_1}\left(\sum_{h:t_h \leqslant Y_i} \frac{\sum_{i=1}^{n_1}\delta_{i,1} I\left(Y_i \geqslant t_{(h)}\right)}{\sum_{i=1}^{n_1} I\left(Y_i \geqslant t_{(h)}\right)}\right) \\
&= \sum_{i=1}^{n_1}\left(\frac{\sum_{i=1}^{n_1}\delta_{i,1}}{n_1}\right) \\
&= \sum_{i=1}^{n_1}\delta_{i,1}
\end{aligned} \tag{9}$$

- Alternatively, the use of a Hazard Ratio or Relative Risk was originally proposed by LeBlanc et al [47, 48]. If the Cox-PH hazard rate estimate is used, then one derives the Log-Hazard Ratio (LHR) statistic, denoted $\hat{\lambda}_{LHR}(m,l)$, for the individuals in both child boxes $B_{g,m,l}$, for $g \in \{1,2\}$, at step (m,l):

$$\begin{aligned}
\hat{\lambda}_{LHR}(m,l) &= \log\left\{\frac{\lambda_0(t)\exp\left[\eta_{1,m,l}\gamma_i(1)\right]}{\lambda_0(t)\exp\left[\eta_{2,m,l}\gamma_i(2)\right]}\right\} \\
&= \log\left\{\frac{\exp(\eta_{1,m,l})}{\exp(0)}\right\} \\
&= \eta_{1,m,l}
\end{aligned} \tag{10}$$

  where $\gamma_i(1) = 1$ and $\gamma_i(2) = 0$ by convention.

Finally, all the above three peeling criteria statistics can be used to maximize the differences in survival outcomes between two consecutive boxes $\hat{B}_{m,l-1}$ and $\hat{B}_{m,l}$ of the box induction/peeling sequence. This leads to the derivation of the corresponding box rate of increase estimate $\hat{r}(m,l)$, at step (m,l), according to equation 6:

$$\hat{r}_{LRT}(m,l) = \frac{\hat{\chi}_{LRT}(m,l) - \hat{\chi}_{LRT}(m,l-1)}{\hat{\beta}_{m,l-1} - \hat{\beta}_{m,l}} \tag{11}$$

$$\hat{r}_{CHS}(m,l) = \frac{\hat{\Lambda}_{CHS}(m,l) - \hat{\Lambda}_{CHS}(m,l-1)}{\hat{\beta}_{m,l-1} - \hat{\beta}_{m,l}} = \frac{\sum_{i=1}^{n_{1,m,l}} \delta_{i,1,m,l} - \sum_{i=1}^{n_{1,m,l}} \delta_{i,1,m,l-1}}{\hat{\beta}_{m,l-1} - \hat{\beta}_{m,l}} \tag{12}$$

$$\hat{r}_{LHR}(m,l) = \frac{\hat{\lambda}_{LHR}(m,l) - \hat{\lambda}_{LHR}(m,l-1)}{\hat{\beta}_{m,l-1} - \hat{\beta}_{m,l}} = \frac{\eta_{1,m,l} - \eta_{1,m,l-1}}{\hat{\beta}_{m,l-1} - \hat{\beta}_{m,l}} \tag{13}$$

2.2.4 Comments

An alternative estimator is to consider the conditional probability $P_{g,m,l}(t|\mathbf{x}_i) = \Pr(T_i \leqslant t|\mathbf{x}_i \in B_{g,m,l})$, which amounts to computing the Nelson-Aalen estimator $\hat{H}_{g,m,l}(t)$ conditioning on the data in a given child box $B_{g,m,l}$. Although this probability is interpretable and estimable, it is by definition a function of an observed event (death) at time t (in a given child box $B_{g,m,l}$). So, one would need to fix a meaningful survival time t. One could think, for instance, of the box median survival time (as is commonly done) or the box maximal event time. In addition, this would induce a likely loss of "power" in contrast to an estimator based on the global survival distribution. As a result, for a given choice of t, this probability may be easy to estimate in some boxes but not estimable in other boxes after sufficient peeling.

Note that the Nelson–Aalen estimator is known to imply conservation of events [56], that is in this case, the total number of deaths is conserved in each child box $B_{g,m,l}$. In fact, that is what the Cumulative Hazard Summary (CHS) statistic amounts to (see eq: 9).

A modified summary statistic derived from the Nelson–Aalen is also possible by normalizing $\hat{\Lambda}_{CHS}(m,l)$ to the total box sample size n_g. This would have the advantage of hedging against large versus small box bias. In our experience, this could be important in the case of discrete covariates.

In addition to the above assumption on the censoring mechanism, the CPH-derived Log-Hazard Ratio or Relative Risk statistic to be used in equation 6 assumes proportional hazards, which may not be realistic. For this reason, this survival peeling criterion is referred to the reader as not preferred and left as a means of comparison to potentially better alternative survival peeling criteria described above (section 2.2.3).

2.2.5 Alternative Survival Peeling Criteria

Further discussion of the use of the above estimators is found in our comparisons of numerical results (section 4.3) and in the discussion (6). Additionally, we mention below a few more alternative survival peeling criteria, although none of these is preferred nor implemented in our R package.

1. It is common to estimate the hazard rate of the simplest parametric survival model (exponential survival model) by the parametric Maximum Likelihood Estimator (MLE). Using above notation and dropping further step subscripts (m,l) for simplicity, if we let $T_i \sim \text{Exp}(\lambda)$ for $i \in \{1,\ldots,n_g\}$ then the parametric MLE is: $\hat{\lambda}_g(t) = \frac{\sum_{i=1}^{n_g} \delta_i \gamma_i(g)}{\sum_{i=1}^{n_g} Y_i \gamma_i(g)}$. The use of this simple parametric estimator of hazard rate for a box was originally proposed by LeBlanc et al. [47, 48]. Since it is always estimable, it could be used to maximize the box rate of increase $r(m,l)$ (eq. 6) at each step (m,l). It also does come with likely the least variance. However, the underlying assumption of constant hazard rate over the duration of time makes it potentially unrealistic and therefore not preferred.
2. The Log-Rank Score Test statistic for splitting in trees (see [37]) or their ensemble versions [40] is another potential criterion available. Note that if there are no tied event times, the Log-Rank Test and the Log-Rank Score Test statistics are identical and, unless there are a large number of tied times, will give very similar results.

Although some may argue that residuals are counter intuitive to use as a peeling criterion, others have used them. For instance, the use of Martingale residuals is strongly recommended by Kehl et al. [43]. Their claim is that they perform better than the deviance residuals, which are a transformation of the Martingale residuals correcting for long tails of the residual distribution. However, others have found that the deviance residuals lead to better rule induction results for bump hunting (Steve Horvath et al.'s personal communication and [50]). Therneau et al [69] have also found that using the deviance residuals as a splitting criterion in regression trees leads to better results than the Martingale residuals.

1. Martingale Residuals $M_i = \delta_i - \hat{\Lambda}(Y_i, \mathbf{x}_i) = \delta_i - \hat{\Lambda}_0(Y_i)\exp(\boldsymbol{\eta}^T\mathbf{x}_i)$, for the ith observation, result from fitting an intercept-only Cox regression to the censored survival times. The idea is to use these as new (uncensored) outcomes in the model instead of time [69], where δ_i is the event indicator and $\hat{\Lambda}_0(Y_i)$ is a non-parametric estimate of the baseline cumulative hazard function for the entire sample.
2. Deviance Residuals $D_i = \text{sign}(\hat{M}_i)\sqrt{2\left[\delta_i \log\left(\frac{\delta_i}{\hat{\Lambda}_0(Y_i)}\right) - \hat{M}_i\right]}$, for the ith observation, have a less symmetric distribution than Martingale residuals [69]. LeBlanc and Crowley [45] also demonstrated that (i) using deviance residuals in regression trees is similar to the survival tree methods presented by Segal [62] and Ciampi et al. [11], and that (ii) using deviance residuals is more efficient than using Martingale residuals with regression trees.

2.2.6 Box End-Point Statistics

Below is a summary of box end-point statistics of interest one can derive in our Survival Bump Hunting method. Each is defined for each step (m, l) and all are implemented in our R package `PRIMsrc` [16]:

1. Log Hazards Ratios (LHR), denoted $\lambda(m, l)$ between the highest-risk group/box and lower-risk groups/boxes of the same generation.
2. Log-Rank Test statistic (LRT), denoted $\chi(m, l)$ between the highest-risk group/box and lower-risk groups/boxes of the same generation.
3. Concordance Error Rate (CER), denoted $\theta(m, l)$ in the highest-risk group/box, that is a prediction performance metric taking censoring into account. For each step (m, l), $\theta(m, l) = 1 - C(m, l)$, where C is Harrel's Concordance Index for censored data [34], a rank correlation U-statistic, to estimate the probability of concordance between predicted and observed survival times.
4. Event-Free Probability (EFP), denoted $P_0(m, l)$ or probability of non-event until a certain time $T(m, l)$ in the highest-risk group/box (Figure 2 left). For instance, the Probability of Event-Free Survival (PEFS) or the Survival Rate (SR) are frequently used. However, $P_0(m, l)$ may not always be reached for a specified time $T(m, l)$. In this case, we determine the limit end-point $P_0'(m, l)$ or Minimal Event-Free Probability ($MEFP$) and corresponding maximal time $T'(m, l)$, which are always observable (see Figure 2 left).
5. Event-Free Time (EFT), denoted $T_0(m, l)$ or time to reach a certain end-point probability $P(m, l)$ in the highest-risk group/box (Figure 2 right). For instance, the Median Survival (MS) is frequently used to indicate the period of time where 50% of subjects have reached survival. However, $T_0(m, l)$ may not always be reached for a certain probability $P(m, l)$. In this case, we determine the limit end-point $T_0'(m, l)$ or Maximal Event-Free Time ($MEFT$) and corresponding minimal probability $P'(m, l)$, which are always observable (see Figure 2 right).
6. Box characteristics:
   - $2p$ box edges $\left[t_j^-(m, l), t_j^+(m, l)\right]_{j=1}^p$,
   - box support (mass) $\beta(m, l)$
   - box membership indicator $\boldsymbol{\gamma}(m, l)$
7. Traces of Covariate Usage $VU(m, l)$ and Covariate Importance $VI(m, l)$
8. Kaplan–Meir curves of survival probability values with log-rank test permutation p-values $p(m, l)$

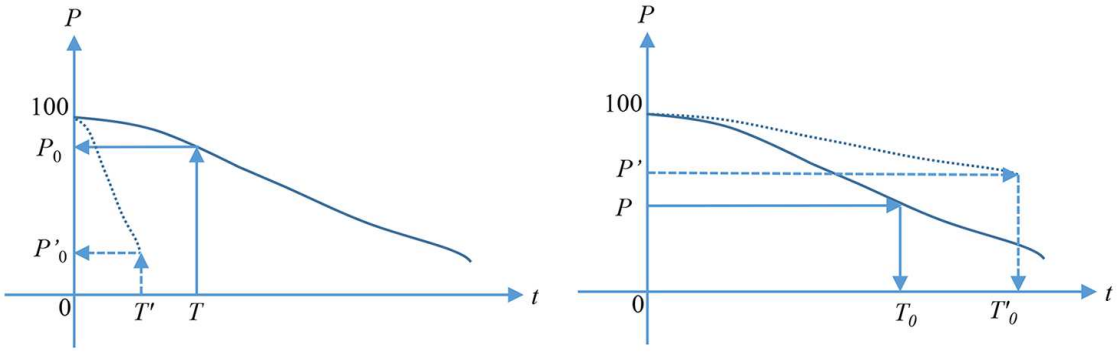


Figure 2: *Survival end points statistics used in Survival Bump Hunting at each step (m, l) of the box generation. Left: Event-Free Probability P_0 and Minimal Event-Free Probability (*MEFP*) P'_0. Right: Event-Free Time T_0 and Maximal Event-Free Time (*MEFT*) T'_0. Subscripts (m, l) are dropped for simplification but understood.*

2.2.7 Estimation by Patient Recursive Survival Peeling

The strategy employed here is a recursive peeling algorithm for survival bump hunting. Our "Patient Recursive Survival Peeling" method proceeds similarly ato which it is done in PRIM except for the box induction peeling/pasting criteria and the induction stopping rule (see section 2.1.4):

Algorithm 1 Patient Recursive Survival Peeling (annotated below w.l.o.g for a maximization problem).

- Start with the training data $\mathcal{L}_{(1)}$ and a maximal box $\hat{B}_1$ containing it
- For $m \in \{1, \ldots, M\}$:
  1: Generate a box $\hat{B}_m$ using the remaining training data $\mathcal{L}_{(m)}$
  2: For $l \in \{1, \ldots, L\}$:
  - Top-down peeling: Generate a box $\hat{B}_{m,l}$ by conducting a stepwise covariate selection/usage: shrink the box by compressing one face (peeling), so as to peel off a quantile α_0 of observations of a covariate $\mathbf{x}_j$ for $j \in \{1, \ldots, p\}$. Choose the direction of peeling j that yields the largest box increase rate $\hat{r}(m, l)$ of the statistic used as peeling criterion between box $\hat{B}_{m,l-1}$ and $B_{m,l}$ in the next generation: Log-Rank Test $\hat{\chi}_{LRT}(m, l)$, Cumulative Hazard Summary $\hat{\Lambda}_{CHS}(m, l)$, Log Hazards Ratio $\hat{\lambda}_{LHR}(m, l)$. The current box $\hat{B}_{m,l-1}$ is then updated: $\hat{B}_{m,l} = \hat{B}_{m,l-1} \backslash \hat{b}^*_{m,l}$, where $\hat{b}^*_{m,l} = \underset{\hat{b}_{m,l} \in C(b_{m,l})}{\text{argmax}} [\hat{r}(m, l)]$
  - Bottom-up pasting: Expand the box along any face (pasting) as long as the resulting box increase rate $\hat{r}(m, l) > 0$
  - Stop the peeling loop until a minimal box support $\hat{\beta}_{m,L}$ of $\hat{B}_{m,L}$ is such that it reached a minimal box support $0 \leqslant \beta_0 \leqslant 1$, expressed as a fraction of the data: $\hat{\beta}_{m,L} \leqslant \beta_0$
  - $l \leftarrow l + 1$
  3: Step #2 give a sequence of nested boxes $\{\hat{B}_{m,l}\}_{l=1}^{L}$, where L is the estimated number of peeling/pasting steps with different numbers of observations in each box. Call the next box $\hat{B}_{m+1} = \hat{B}_{m,L}$. Remove the data in box $\hat{B}_m$ from the training data: $\mathcal{L}_{(m+1)} = \mathcal{L}_{(m)} \backslash \hat{B}_m$
  4: Stop the covering loop when running out of data or when a minimal number of observations remains within the last box $\hat{B}_M$, say $\hat{\beta}_M \leqslant \beta_0$
  5: $m \leftarrow m + 1$
- Steps #1 – #5 produce a sequence of (not necessarily nested) boxes $\{\hat{B}_m\}_{m=1}^{M}$, where M is the estimated total number of boxes covering $\mathcal{L}_{(1)}$
- Collect the decision rules of all boxes $\{\hat{B}_m\}_{m=1}^{M}$ into a simple final decision rule $\hat{\mathcal{R}}$ of the solution region $\hat{R}$ of the form: $\hat{\mathcal{R}} = \bigcup_{m=1}^{M} \hat{\mathcal{R}}_m$, where $\hat{\mathcal{R}}_m = \bigcap_{j \in J} (\mathbf{x}_j \in [t^-_{j,m}, t^+_{j,m}])$ giving a full description of the estimated bumps in the entire input space

3 Cross-Validation for Recursive Peeling Methods and a Survival/Risk Outcome

3.1 Split-Sample-Validation

3.1.1 Setup

We previously tested the possibility of finding survival bumps in a small dataset, namely the Veteran's Administration lung cancer trial data from Kalbfleisch and Prentice [42]. We could unravel interesting subgroups of patients with a poor survival time that could be characterized by a set of descriptive rules on the predictors including treatment intervention. Typically, this was indicative that an alternative intervention therapy could be required for these non-responders. While this approach showed promising results, it remained naive in that possible issues of bias and overfitting were not kept in check by model validation.

Assessment of model performance (e.g. prediction accuracy) requires the use of separation of the whole data $\mathcal{L}$ between a "training set" $\mathcal{L}^{\setminus t}$ used to build a model and an independent "testing set" $\mathcal{L}^t$ used to assess model performance. To do so, the Split-Sample Validation technique (a.k.a Full Validation) is possible. Using this approach, a model is entirely developed on the training set $\mathcal{L}^{\setminus t}$. Then, samples in the independent testing set $\mathcal{L}^t$ are used to determine the error rates. The samples in the testing set are never to be used for any aspect of model development such as variable selection and calibration and can therefore be used to check model performance [55, 71].

Here, cross-validation of box estimates should include all steps of the box generation sequence $\{B_m\}_{m=1}^M$ i.e. for the (outer) coverage loop of our "Patient Recursive Survival Peeling" method (Algorithm 1), each step of which involves a peeling sequence $\{B_{m,l}\}_{l=1}^L$ of the (inner) box peeling/induction loop. However, for simplicity, cross-validation designs of box estimates and resulting decision rule $\hat{\mathcal{R}}_m$ are shown below for *fixed* $m \in \{1, \ldots, M\}$ of the complete box sequence $\{\{\hat{B}_{m,l}\}_{l=1}^L\}_{m=1}^M$, so that subscript m is further dropped. Without loss of generality, fix $m = 1$ (first coverage box).

3.1.2 Estimated Box Quantities of Interest

Using previous notation, if we let $\hat{B}_l$ be the lth trained box and $\hat{\beta}_l$ be its estimated box support for $l \in \{1, \ldots, L\}$ of a box peeling sequence $\{\hat{B}_l\}_{l=1}^L$, then the test-set mean estimate of a box quantity of interest q for the lth peeling step is indexed by the lth test box support $\hat{\beta}_l^t$ as follows:

$$q(\hat{\beta}_l^t) = \frac{1}{n^t \hat{\beta}_l^t} \sum_{i=1}^{n^t} \hat{q}_i^t I\left(\mathbf{x}_i^t \in \hat{B}_l\right) \tag{14}$$

where $q(\cdot)$ is the functional corresponding to the quantity q, $\hat{\beta}_l^t = \frac{1}{n^t} \sum_{i=1}^{n^t} I\left(\mathbf{x}_i^t \in \hat{B}_l^t\right)$ and $\hat{q}_i^t, \mathbf{x}_i^t, n^t$ are test-set quantities. Useful test-set quantities for the highest-risk box are box end-point statistics mentioned in section 2.2.6.

3.2 K-fold Cross-Validation

3.2.1 Resampling Design - Notation

Although using a fully independent test set for evaluating a predictive bump hunting model is always advisable, the sample size n in discovery-based studies is often too small to effectively split the data into training and testing sets and provide accurate estimates [6, 22, 64]. In such cases, resampling techniques such as K-fold Cross-Validation (CV) are required [2, 55].

In resampling based on full K-fold cross-validation, the whole data $\mathcal{L}$ is randomly partitioned into K approximately equal parts of test samples or test-sets $(\mathcal{L}_1, \ldots, \mathcal{L}_k, \ldots, \mathcal{L}_K)$. For each test-set $\mathcal{L}_k$, for $k \in \{1, \ldots, K\}$, a training set $\mathcal{L}_{(k)}$ is formed from the union of the remaining $K - 1$ subsets: $\mathcal{L}_{(k)} = \mathcal{L} \setminus \mathcal{L}_k$. The process is repeated K times, so that K test-sets $\mathcal{L}_k$ are formed of about equal size and K corresponding training subsets $\mathcal{L}_{(k)}$, for $k \in \{1, \ldots, K\}$. Typically, $K \in \{3, \ldots, 10\}$. The training samples are approximately of size $\approx n(K-1)/K$ and the test samples are of size $n^t \approx n/K$.

3.2.2 Cross-Validation Techniques

Recently, we described a cross-validation technique for recursive peeling methods in a survival/risk setting [14]. The subject of this section is to give a more in-depth development of this strategy and compare it to standard cross-validation techniques.

There are issues when dealing with K-fold CV: first, how to cross-validate a simple peeling trajectory $\{\hat{B}_l\}_{l=1}^{L}$ and related statistics is not straightforward; second, how to cross-validate survival curve estimates and related statistics is also not intuitive (see also [65]); third, the data splitting in the cross-validation step should balance the class distributions of the outcome (i.e. here the censoring rate) within the cross-validation splits, which we call "stratified random splitting by conservation of events". So, regular K-fold cross-validation is not directly applicable to the joint task of box decision rules making by recursive peeling and survival estimation. One must design a specific cross-validation technique(s) of survival bump hunting that is amenable to this joint task.

Hence, we propose two techniques by which K-fold cross-validation estimates can be computed:

- *Averaging Technique*: Estimations are first computed for each "in-box" test subset samples, then averaged over the cross-validation loops of random splitting to give the "Averaged Cross-Validation" estimates (see details in section below 3.3).
- *Combining Technique*: All "in-box" test subset samples are first collected from all the cross-validation loops of random splitting to build a *combined* test "in-box" and corresponding *combined* test "in-box" samples to compute *once* the final "Combined Cross-Validation" estimates (see details in section below 3.4).

Note that, unlike in the averaging technique, cross-validated combined estimates are computed on test samples of size n instead of $n^t \approx n/K$, which could be an advantage in the case of tiny sample size n. In our numerical analyses, both strategies were compared with each other and with the situation of no cross-validation (see result section 4.3).

Finally, to account for the high variability of cross-validated estimates [22, 24, 51], we iterate each cross-validation procedure several times over some replicates B (typically, $B \geqslant 10$) to average the estimates and reduce their variance. This so-called "Replicated Cross-Validation" approach is further detailed in section below (3.6). Also, aside the Split-Sample-Validation, mentioned in the previous section (3.1), other resampling techniques are available, which we left for discussion (section 6).

3.2.3 Model Peeling Length Optimization Criterion

In model tuning, a trade-off between under-fitting and over-fitting can be achieved by optimizing an empirical function or objective criterion that takes censoring into account using cross-validation. The "optimization criterion" that we derive below is adapted to the task of of fitting a survival bump hunting model by recursive peeling with a survival outcome. Specifically, we tune a peeling model by optimizing its complexity, that is, the final length or number of peeling steps L of the peeling sequence. The reason is that, for a given set of variables/covariates, the final length L of a peeling model only depends on the peeling meta-parameters α_0 (assumed fixed here) and β_0 (see section 2.1.4). In fact, an upper bound on the length of all possible peeling trajectories is given by $L_{\alpha_0,\beta_0} = \left\lceil \frac{\log(\beta_0)}{\log(1-\alpha_0)} \right\rceil$ (see [31] for details). So, a cheaper cross-validation can be achieved on L only rather than on α_0 and β_0 simultaneously.

Assuming m fixed (see step #2 of Algorithm 1) and dropping subscript m for simplification, the process of model building is repeated K times for $k \in \{1, \ldots, K\}$ as follows. First, let $l(k)$ denote the lth peeling step in the kth trajectory for $l \in \{1, \ldots, L(k)\}$ and $k \in \{1, \ldots, K\}$, where $L(k)$ denotes the final length of a trained peeling model. Note that $L(k) \leqslant L_{\alpha_0,\beta_0}$, for all $k \in \{1, \ldots, K\}$, but, in general this inequality is strict for large enough sample sizes. Let $\hat{B}_{l(k)}$ be the trained box of support $\hat{\beta}_{l(k)}$ of the box peeling sequence $\{\hat{B}_{l(k)}\}_{l=1}^{L}$ that is constructed from training set $\mathcal{L}_{(k)}$, leaving out the test-set $\mathcal{L}_k$ during all aspects of model building including covariate selection.

Second, once a resulting trained decision rule, abbreviated $\mathcal{R}_k$, and box definition estimates are generated from each training set $\mathcal{L}_{(k)}$, cross-validated estimates of box end-points statistics (described in 2.2.6)

are computed using the left-out test-set $\mathcal{L}_k$. Three of these are the Log Hazard Ratio (LHR), Log-Rank Test (LRT) and the cross-validated estimate of prediction performance, namely the Concordance Error Rate (CER) that is obtained by calculating the test-set error rate using the left-out test-set $\mathcal{L}_k$.

In the subsequent sections, we denote by superscript cv any cross-validated estimate on the test-set $\mathcal{L}_k$. Since peeling lengths $L(k)$ are not necessarily equal for all $k \in \{1, \ldots, K\}$, we use the following cross-validated maximum peeling length $\hat{L}_m^{cv}$ over the K trajectories:

$$\hat{L}_m^{cv} = \min_{k \in \{1,\ldots,K\}} [L(k)] \tag{15}$$

After K rounds of training and testing are complete and (averaged or combined) test profiles of LHR, LRT or CER estimates are determined for each step $l \in \{1, \ldots, \hat{L}_m^{cv}\}$, model tuning is done by determining the optimal peeling length $\hat{L}^{cv}$ of the peeling sequence. To that end, one uses the maximization of the (averaged or combined) test profiles of LHR and LRT or the minimization of the (averaged or combined) test profile of CER as criterion. Formally:

$$\hat{L}^{cv} = \underset{l \in \{1,\ldots,\hat{L}_m^{cv}\}}{\operatorname{argmax}} \left[\hat{\lambda}^{cv}(l)\right] \quad \text{or} \quad \hat{L}^{cv} = \underset{l \in \{1,\ldots,\hat{L}_m^{cv}\}}{\operatorname{argmax}} \left[\hat{\chi}^{cv}(l)\right] \quad \text{or} \quad \hat{L}^{cv} = \underset{l \in \{1,\ldots,\hat{L}_m^{cv}\}}{\operatorname{argmin}} \left[\hat{\theta}^{cv}(l)\right], \tag{16}$$

where $\hat{\lambda}^{cv}(l)$ is the cross-validated LHR) in the high-risk box at step l, $\hat{\chi}^{cv}(l)$ is the cross-validated LRT between the high vs. low-risk box at step l and $\hat{\theta}^{cv}(l)$ is the cross-validated CER between high-risk box predicted and observed survival times at step l.

Depending on the desired degree of conservativeness, the usual one-standard-error rule [36] may be applied in combination with the profiles minimizer or maximizer to get smaller estimates corresponding to one standard-error below the maximum of LHR and LRT or standard-error above the minimum of CER. In the subsequent sections, we denote by superscript cv any cross-validated estimate on the test-set $\mathcal{L}_k$.

3.3 K-fold Averaged Cross-Validation

In K-fold Averaged Cross-Validation, the *averaged* cross-validated estimate of a box quantity q at the $l(k)$th step of the box peeling sequence is based on the test samples falling within the trained box $\hat{B}_{l(k)}$. The *averaged* cross-validated estimate of q at step l is simply computed by averaging the estimates obtained from all test boxes computed over all K cross-validation loops. Specifically, each test-set $\mathcal{L}_k$ is used to estimate the $l(k)$th test box membership indicator $\hat{\gamma}^t_{l(k)}$ from the model grown on the training set $\mathcal{L}_{(k)}$. The corresponding test box support $\hat{\beta}^t_{l(k)}$ is directly derived from $\hat{\gamma}^t_{l(k)}$ by computing the fraction of test data falling within the trained box $\hat{B}_{l(k)}$. The $l(k)$th estimate of the box quantity q is indexed by the corresponding test box support $\hat{\beta}^t_{l(k)}$. For each training set $\mathcal{L}_{(k)}$, a trajectory curve $q(x)$ of a box quantity q is defined as a piecewise constant curve, evaluated at the $l(k)$th test box support $\hat{\beta}^t_{l(k)}$, so that each trajectory curve is: $q(x) = q(\hat{\beta}^t_{l(k)})$ for $\hat{\beta}^t_{l(k)+1} \leqslant x \leqslant \hat{\beta}^t_{l(k)}$ (Figure 3), where $q(\hat{\beta}^t_{l(k)})$ is derived as in equation 14. The *averaged* CV trajectory curve $\hat{q}^{cv}(x)$ of length $\hat{L}_m^{cv}$ is simply the average of the K trajectory curves over the K cross-validation loops: $\hat{q}^{cv}(x) = \frac{1}{K}\sum_{k=1}^{K} q(\hat{\beta}^t_{l(k)})$ for $\hat{\beta}^t_{l(k)+1} \leqslant x \leqslant \hat{\beta}^t_{l(k)}$.

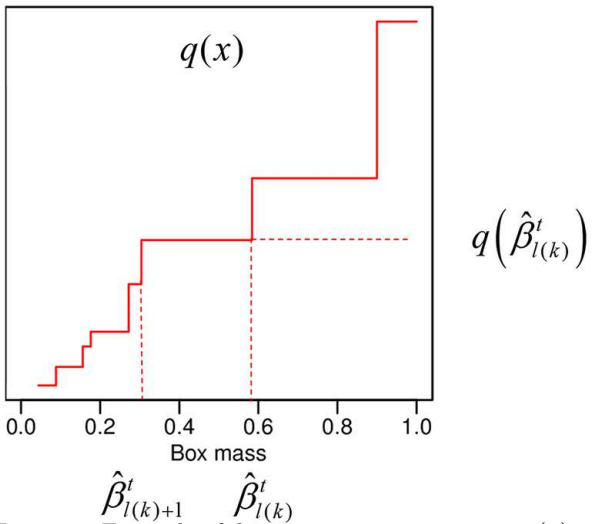


Figure 3: *Example of decreasing trajectory curve $q(x)$ of a box quantity q. Notice the piecewise constant curve of $q(x)$ for $\hat{\beta}^t_{l(k)+1} \leqslant x \leqslant \hat{\beta}^t_{l(k)}$. By convention, we define $\hat{\beta}^t_0 = 1$, $\hat{\beta}^t_{L(k)} = \beta_0$ and $\hat{\beta}^t_{L(k)+1} = 0$.*

Formally, we show below how things are computed from an initial set of K trained peeling trajectories:

For the kth training set $\mathcal{L}_{(k)}$:	First peeling step in the kth trajectory		direction of peeling $\rightarrow$		Last peeling step in the kth trajectory
kth training trajectory with box definitions $\rightarrow$	$1(k)$: $\hat{B}_{1(k)}$	$\cdots$	$l(k)$: $\hat{B}_{l(k)}$	$\cdots$	$L(k)$: $\hat{B}_{L(k)}$
kth test box membership indicators $\rightarrow$	$\hat{\boldsymbol{\gamma}}^{t\ T}_{1(k)} = \left[\hat{\gamma}^t_{i,1(k)}\right]_{i=1}^{n^t} = \left[I[\mathbf{x}^t_i \in \hat{B}_{1(k)}]\right]_{i=1}^{n^t}$	$\cdots$	$\hat{\boldsymbol{\gamma}}^{t\ T}_{l(k)} = \left[\hat{\gamma}^t_{i,l(k)}\right]_{i=1}^{n^t} = \left[I[\mathbf{x}^t_i \in \hat{B}_{l(k)}]\right]_{i=1}^{n^t}$	$\cdots$	$\hat{\boldsymbol{\gamma}}^{t\ T}_{L(k))} = \left[\hat{\gamma}^t_{i,L(k)}\right]_{i=1}^{n^t} = \left[I[\mathbf{x}^t_i \in \hat{B}_{L(k)}]\right]_{i=1}^{n^t}$
kth test box supports $\rightarrow$	$\hat{\beta}^t_{1(k)} = \frac{1}{n^t}\sum_{i=1}^{n^t} \hat{\gamma}^t_{i,1(k)}$	$\cdots$	$\hat{\beta}^t_{l(k)} = \frac{1}{n^t}\sum_{i=1}^{n^t} \hat{\gamma}^t_{i,l(k)}$	$\cdots$	$\hat{\beta}^t_{L(k)} = \frac{1}{n^t}\sum_{i=1}^{n^t} \hat{\gamma}^t_{i,L(k)}$
kth test box estimated quantities $\rightarrow$	$q\left(\hat{\beta}^t_{1(k)}\right)$	$\cdots$	$q\left(\hat{\beta}^t_{l(k)}\right)$	$\cdots$	$q\left(\hat{\beta}^t_{L(k)}\right)$
Averaged CV test box quantities over K trajectories $\rightarrow$	$\hat{q}^{cv}(1) = \frac{1}{K}\sum_{k=1}^{K} q\left(\hat{\beta}^t_{1(k)}\right)$	$\cdots$	$\hat{q}^{cv}(l) = \frac{1}{K}\sum_{k=1}^{K} q\left(\hat{\beta}^t_{l(k)}\right)$	$\cdots$	$\hat{q}^{cv}(\hat{L}^{cv}_m) = \frac{1}{K}\sum_{k=1}^{K} q\left(\hat{\beta}^t_{\hat{L}^{cv}_m}\right)$

From the K test trajectories, one derives first the "Averaged CV" optimal peeling length of the peeling trajectory, according to the optimization criterion for model selection as in equation 16:

$$\hat{L}^{cv} = \underset{l\in\{1,\ldots,\hat{L}^{cv}_m\}}{\text{argmax}} \left[\hat{\lambda}^{cv}(l)\right] \quad \text{or} \quad \hat{L}^{cv} = \underset{l\in\{1,\ldots,\hat{L}^{cv}_m\}}{\text{argmax}} \left[\hat{\chi}^{cv}(l)\right] \quad \text{or} \quad \hat{L}^{cv} = \underset{l\in\{1,\ldots,\hat{L}^{cv}_m\}}{\text{argmin}} \left[\hat{\theta}^{cv}(l)\right]$$

Then, one derives "Averaged CV" estimates for each step $l \in \{1, \ldots, \hat{L}^{cv}\}$ as follows:

- The "Averaged CV" box definition, which can be written as the outer product of $|J|$ intervals as in equation 3, is formed by taking the rectangular box where each of the $2|J|$ edge is averaged over the K cross-validation loops:

$$\hat{B}^{cv}(l) = \bigtimes_{j\in J} [\hat{t}^-_{j,l}, \hat{t}^+_{j,l}] \quad \text{where for each } j \in J, \quad \begin{cases} \hat{t}^-_{j,l} = \underset{k\in\{1,\ldots,K\}}{\text{ave}} \ [\hat{t}^-_{j,l(k)}] \\ \hat{t}^+_{j,l} = \underset{k\in\{1,\ldots,K\}}{\text{ave}} \ [\hat{t}^+_{j,l(k)}] \end{cases}$$

- The "Averaged CV" box membership indicator is formed by counting the data within the "Averaged CV" box:

$$\hat{\boldsymbol{\gamma}}^{cv\ T}(l) = \left[I[\mathbf{x}_i \in \hat{B}^{cv}(l)]\right]_{i=1}^{n}$$

- The "Averaged CV" box support is computed as the fraction of data within the "Averaged CV" box:

$$\hat{\beta}^{cv}(l) = \frac{1}{n}\sum_{i=1}^{n} I[\mathbf{x}_i \in \hat{B}^{cv}(l)]$$

- The "Averaged CV" box end-point quantity q is taken as the averaged CV trajectory curve evaluated at the $l(k)$th test box support $\hat{\beta}^t_{l(k)}$:

$$\hat{q}^{cv}(l) = \frac{1}{K}\sum_{k=1}^{K} q\left(\hat{\beta}^t_{l(k)}\right) \text{ , where}$$

$$q\left(\hat{\beta}^t_{l(k)}\right) = \frac{1}{n^t \hat{\beta}^t_{l(k)}}\sum_{i=1}^{n^t} \hat{q}^t_i I\left(\mathbf{x}^t_i \in \hat{B}_{l(k)}\right) \text{ as in equation 14.}$$

The latter is done for the "Averaged CV" box estimates of: (i) The Log Hazard Ratio (LHR) in the high-risk box: $\hat{\lambda}^{cv}(l) = \frac{1}{K}\sum_{k=1}^{K}\lambda\left(\hat{\beta}^t_{l(k)}\right)$; (ii) The Log-Rank Test (LRT) between the high vs. low-risk box: $\hat{\chi}^{cv}(l) = \frac{1}{K}\sum_{k=1}^{K}\chi\left(\hat{\beta}^t_{l(k)}\right)$; (iii) The Concordance Error Rate (CER) between high-risk box predicted and observed survival times: $\hat{\theta}^{cv}(l) = \frac{1}{K}\sum_{k=1}^{K}\theta\left(\hat{\beta}^t_{l(k)}\right)$; (iv) The Minimal Event-Free Probability ($MEFP$): $\widehat{P'_0}^{cv}(l) = \frac{1}{K}\sum_{k=1}^{K} P'_0\left(\hat{\beta}^t_{l(k)}\right)$; (v) The Minimal Event-Free Time ($MEFT$): $\widehat{T'_0}^{cv}(l) = \frac{1}{K}\sum_{k=1}^{K} T'_0\left(\hat{\beta}^t_{l(k)}\right)$.

3.4 K-fold Combined Cross-Validation

In K-fold Combined Cross-Validation, for each loop, samples from the training set are used to train a peeling model of a certain length, then samples from the test-set are used to determine the "in-box" test-set samples falling into the trained box. Eventually, all "in-box" test samples are combined together and all "out-of-box" test samples are combined together as well. So, in K-fold Combined CV, estimate of a box quantity q is computed *once* on the collective test-set "in-box" samples, formed over the K cross-validation loops. This allows the estimation of box quantities and box survival distribution curves for both "in-box" and "out-of-box" samples.

Specifically, each test-set $\mathcal{L}_k$ is used to estimate the test box membership indicator $\tilde{\gamma}^t_{l(k)}$ from the model grown on the kth training set $\mathcal{L}_{(k)}$. The lth *combined* cross-validated test box membership indicator $\tilde{\boldsymbol{\gamma}}^{cv}(l)$ is formed *once* by taking the vector concatenation of all the cross-validated test box membership indicators $\{\tilde{\gamma}^t_{l(k)}\}_{k=1}^{K}$ over the K cross-validation loops. The corresponding lth combined cross-validated test box support $\tilde{\beta}^{cv}(l)$ is then directly derived from $\tilde{\boldsymbol{\gamma}}^{cv}(l)$.

The *combined* cross-validated estimate of a box quantity q at the lth step of the peeling trajectory is then computed *once* from the combined cross-validated test box membership indicator $\tilde{\boldsymbol{\gamma}}(l)^{cv}$ and indexed by the corresponding test box support $\tilde{\beta}(l)^{cv}$. Here, the *combined* cross-validated trajectory curve $\tilde{q}_k(x)$ is defined as the piecewise constant curve of length $\tilde{L}^{cv}_m$, evaluated at the lth combined cross-validated test box membership indicator $\tilde{\boldsymbol{\gamma}}^{cv}(l)$.

Formally, we show below how things are computed from an initial set of K trained peeling trajectories (where || denotes the concatenation operator).

For the kth training set $\mathcal{L}_{(k)}$:	First peeling step in the kth trajectory		direction of peeling $\rightarrow$		Last peeling step in the kth trajectory
kth training trajectory with box definitions $\rightarrow$	$1(k)$ $\boxed{\tilde{B}_{1(k)}}$	$\cdots$	$l(k)$ $\boxed{\tilde{B}_{l(k)}}$	$\cdots$	$L(k)$ $\boxed{\tilde{B}_{L(k)}}$
kth test box membership indicators $\rightarrow$	$\tilde{\boldsymbol{\gamma}}^{t\ T}_{1(k)} = \left[I[\mathbf{x}^t_i \in \tilde{B}_{1(k)}] \right]^{n^t}_{i=1}$	$\cdots$	$\tilde{\boldsymbol{\gamma}}^{t\ T}_{l(k)} = \left[I[\mathbf{x}^t_i \in \tilde{B}_{l(k)}] \right]^{n^t}_{i=1}$	$\cdots$	$\tilde{\boldsymbol{\gamma}}^{t\ T}_{L(k)} = \left[I[\mathbf{x}^t_i \in \tilde{B}_{L(k)}] \right]^{n^t}_{i=1}$
Combined CV test box membership indicators over K trajectories $\rightarrow$	$\tilde{\boldsymbol{\gamma}}^{cv}(1) = [\tilde{\gamma}^{cv}_i(1)]^n_{i=1} = \Big\Vert_{k=1}^{K} \tilde{\boldsymbol{\gamma}}^t_{1(k)}$	$\cdots$	$\tilde{\boldsymbol{\gamma}}^{cv}(l) = [\tilde{\gamma}^{cv}_i(l)]^n_{i=1} = \Big\Vert_{k=1}^{K} \tilde{\boldsymbol{\gamma}}^t_{l(k)}$	$\cdots$	$\tilde{\boldsymbol{\gamma}}^{cv}(\tilde{L}^{cv}_m) = \left[\tilde{\gamma}^{cv}_i(\tilde{L}^{cv}_m) \right]^n_{i=1} = \Big\Vert_{k=1}^{K} \tilde{\boldsymbol{\gamma}}^t_{\tilde{L}^{cv}_m}$
Combined CV test box supports over K trajectories $\rightarrow$	$\tilde{\beta}^{cv}(1) = \frac{1}{n} \sum_{i=1}^{n} \tilde{\gamma}^{cv}_i(1)$	$\cdots$	$\tilde{\beta}^{cv}(l) = \frac{1}{n} \sum_{i=1}^{n} \tilde{\gamma}^{cv}_i(l)$	$\cdots$	$\tilde{\beta}^{cv}(\tilde{L}^{cv}_m) = \frac{1}{n} \sum_{i=1}^{n} \tilde{\gamma}^{cv}_i(\tilde{L}^{cv}_m)$
Combined CV test box quantities over K trajectories $\rightarrow$	$\tilde{q}^{cv}(1) = q\left[\tilde{\boldsymbol{\beta}}^{cv}(1) \right]$	$\cdots$	$\tilde{q}^{cv}(l) = q\left[\tilde{\boldsymbol{\beta}}^{cv}(l) \right]$	$\cdots$	$\tilde{q}^{cv}(\tilde{L}^{cv}_m) = q\left[\tilde{\boldsymbol{\beta}}^{cv}(\tilde{L}^{cv}_m) \right]$

From the K test trajectories, one derives first the "Combined CV" optimal peeling length of the peeling trajectory, according to the optimization criterion as in equation 16:

$$\tilde{L}^{cv} = \underset{l \in \{1,\ldots,\tilde{L}^{cv}_m\}}{\operatorname{argmax}} \left[\tilde{\lambda}^{cv}(l) \right] \quad \text{or} \quad \tilde{L}^{cv} = \underset{l \in \{1,\ldots,\tilde{L}^{cv}_m\}}{\operatorname{argmax}} \left[\tilde{\chi}^{cv}(l) \right] \quad \text{or} \quad \tilde{L}^{cv} = \underset{l \in \{1,\ldots,\tilde{L}^{cv}_m\}}{\operatorname{argmin}} \left[\tilde{\theta}^{cv}(l) \right]$$

Likewise, from the K test trajectories, one derives "Combined CV" estimates for each step $l \in \{1, \ldots, \tilde{L}^{cv}\}$ as follows:

- The "Combined CV" box membership indicator (Boolean n-vector) is formed by vector-concatenation of all the test box membership indicators over the K cross-validation loops:

$$\tilde{\boldsymbol{\gamma}}^{cv\ T}(l) = [\tilde{\gamma}^{cv}_i(l)]^n_{i=1} = \Big\Vert_{k=1}^{K} \tilde{\boldsymbol{\gamma}}^t_{l(k)} = \Big\Vert_{k=1}^{K} \left[I[\mathbf{x}^t_i \in \tilde{B}_{l(k)}] \right]^{n^t}_{i=1}$$

- The "Combined CV" box definition, which can be written as the outer product of $|J|$ intervals as in equation 3, is formed by taking the rectangular box ($2|J|$ edges) circumscribing all the "in-box" test samples over the K cross-validation loops:

$$\tilde{B}^{cv}(l) = \bigtimes_{j \in J} [\tilde{t}^-_{j,l}, \tilde{t}^+_{j,l}] \quad \text{where for each } j \in J, \quad \begin{cases} \tilde{t}^-_{j,l} = \min\limits_{k \in \{1,\ldots,K\}} [x^t_{i,j}, i \in \{1, \ldots, n^t\} \colon x^t_{i,j} \in \tilde{B}_{l(k)}] \\ \tilde{t}^+_{j,l} = \max\limits_{k \in \{1,\ldots,K\}} [x^t_{i,j}, i \in \{1, \ldots, n^t\} \colon x^t_{i,j} \in \tilde{B}_{l(k)}] \end{cases}$$

- The "Combined CV" box support is computed as the fraction of data within the "Combined CV" box:

$$\tilde{\beta}^{cv}(l) = \frac{1}{n}\sum_{i=1}^{n}\tilde{\gamma}_i^{cv}(l)$$

- The "Combined CV" box end-point quantity q is taken as the result of the functional $q(\cdot)$ evaluated at the lth "Combined CV" test box support $\tilde{\beta}^{cv}(l)$:

$$\tilde{q}^{cv}(l) = q\left[\tilde{\boldsymbol{\beta}}^{cv}(l)\right]$$

The latter is done for the "Combined CV" box estimates of: (i) The Log Hazard Ratio (LHR) in the high-risk box: $\tilde{\lambda}^{cv}(l) = \lambda\left[\tilde{\boldsymbol{\beta}}^{cv}(l)\right]$; (ii) The Log-Rank Test (LRT) between the high vs. low-risk box: $\tilde{\chi}^{cv}(l) = \chi\left[\tilde{\boldsymbol{\beta}}^{cv}(l)\right]$; (ii) The Concordance Error Rate (CER) between high-risk box predicted and observed survival times: $\tilde{\theta}^{cv}(l) = \theta\left[\tilde{\boldsymbol{\beta}}^{cv}(l)\right]$; (iv) The Minimal Event-Free Probability ($MEFP$): $\widetilde{P_0'}^{cv}(l) = P_0'\left[\tilde{\boldsymbol{\beta}}^{cv}(l)\right]$; (v) The Minimal Event-Free Time ($MEFT$): $\widetilde{T_0'}^{cv}(l) = T_0'\left[\tilde{\boldsymbol{\beta}}^{cv}(l)\right]$.

3.5 K-fold Cross-Validation of P-Values

The log-rank test statistic (e.g. χ_1^2 for a two group comparison) is a classical measure to evaluate the statistical significance of separation between survival curves. However, the null distribution of the log-rank test is not valid for cross-validated curves because the observations used to cross-validate the curves are not independent anymore.

For each step $l \in \{1, \ldots, \tilde{L}^{rcv}\}$, we generate the null distribution of the cross-validated log-rank statistic $\tilde{\chi}^{cv(a)}(l)$ for $a \in \{1, \ldots, A\}$ by randomly permuting the correspondence of survival times and censoring indicators of the data and by computing the corresponding cross-validated survival curves and cross-validated log-rank statistic for that permutation. By repeating A times the entire K-fold cross-validation process for many random permutations (typically $A = 1000$), one generates a null distribution of the permuted log-rank statistics (annotated below w.l.o.g. for the case of "Combined CV"):

$$\{\tilde{\chi}^{cv(a)}(l)\}_{a=1}^{A}$$

The proportion of replicates with log-rank statistic greater than or equal to the observed statistic $\tilde{\chi}^{cv}(l)$ for the un-permuted data is the statistical significance level for the test. Log-rank test permutation p-values are then calculated for each step $l \in \{1, \ldots, \tilde{L}^{rcv}\}$ as:

$$\tilde{p}^{cv}(l) = \frac{1}{A}\sum_{a=1}^{A} I\left[\tilde{\chi}^{cv(a)}(l) \geqslant \tilde{\chi}^{cv}(l)\right] \tag{17}$$

These p-values may be discrete: the precision depends on the number A of random permutations and the lower bound $1/A$ may be reached in practise.

3.6 Replicated K-fold Cross-Validation

Typically, K-fold cross-validation is repeated $B = 10 - 100$ times and resulting replicated cross-validated estimates are somehow "averaged" over the replicates. We denote these by the superscript rcv and each replicate by the superscript $cv(b)$, for $b \in \{1, \ldots, B\}$. This is done for either cross-validation technique (shown below w.l.o.g. for the case of "Combined CV").

Formally, one first derives the "Replicated CV" maximal peeling length of the peeling model from the cross-validation replicates, denoted $\bar{L}_m^{rcv}$. To do so, one uses the cross-validated maximum peeling length

$\hat{L}_m^{cv(b)}$ of the peeling trajectory, defined in equation 15, for $b \in \{1, \ldots, B\}$. Formally, the "Replicated CV" maximal peeling length $\bar{L}_m^{rcv}$ is calculated as the ceiling-mean of the cross-validated quantities $\hat{L}_m^{cv(b)}$:

$$\bar{L}_m^{rcv} = \left\lceil \frac{1}{B} \sum_{b=1}^{B} \hat{L}_m^{cv(b)} \right\rceil \tag{18}$$

Next, depending on the optimization criterion used, one gets the "Replicated CV" optimal length of the peeling trajectory:

$$\bar{L}^{rcv} = \underset{l \in \{1,\ldots,\bar{L}_m^{rcv}\}}{\operatorname{argmax}} \left[\bar{\lambda}^{rcv}(l)\right] \quad \text{or} \quad \bar{L}^{rcv} = \underset{l \in \{1,\ldots,\bar{L}_m^{rcv}\}}{\operatorname{argmax}} \left[\bar{\chi}^{rcv}(l)\right] \quad \text{or} \quad \bar{L}^{rcv} = \underset{l \in \{1,\ldots,\bar{L}_m^{rcv}\}}{\operatorname{argmin}} \left[\bar{\theta}^{rcv}(l)\right] \tag{19}$$

where each optimization criterion: (i) The Log Hazard Ratio (LHR) in the high-risk box: $\bar{\lambda}^{rcv}(l)$, (ii) The Log-Rank Test (LRT) between the high vs. low-risk box: $\bar{\chi}^{rcv}(l)$, and (iii) The Concordance Error Rate (CER) between high-risk box predicted and observed survival times: $\bar{\theta}^{rcv}(l)$ is taken as the average estimate over the B replicates for each step $l \in \{1, \ldots, \bar{L}_m^{rcv}\}$ as follows:

$$\bar{\lambda}^{rcv}(l) = \frac{1}{B} \sum_{b=1}^{B} \tilde{\lambda}^{cv(b)}(l) \quad \text{or} \quad \bar{\chi}^{rcv}(l) = \frac{1}{B} \sum_{b=1}^{B} \tilde{\chi}^{cv(b)}(l) \quad \text{or} \quad \bar{\theta}^{rcv}(l) = \frac{1}{B} \sum_{b=1}^{B} \tilde{\theta}^{cv(b)}(l) \tag{20}$$

Using the above "Replicated CV" optimal length of the peeling trajectory, one finally derives "Replicated CV" box end points from the B replicates for each step $l \in \{1, \ldots, \bar{L}^{rcv}\}$ as follows:

- The "Replicated CV" box definition ($2|J|$ edges) is taken as the average-box over the B replicates:

$$\bar{B}^{rcv}(l) = \underset{b \in \{1,\ldots,B\}}{\text{ave}} \left[\tilde{B}^{cv(b)}(l)\right] \tag{21}$$

  where ave$(\cdot)$ denotes the averaging function by edge or dimension j for $j \in J$.

- The "Replicated CV" box membership indicator (Boolean n-vector) is taken as the average-box membership indicator, observed to be nearly equal to the point-wise majority vote over the B replicates:

$$\bar{\boldsymbol{\gamma}}^{rcv}(l) = \left[I[\mathbf{x}_i \in \bar{B}^{rcv}(l)]\right]_{i=1}^{n} \approx \left[I\left(\sum_{b=1}^{B} \tilde{\gamma}_i^{cv(b)}(l) \geqslant \left\lceil \frac{B}{2} \right\rceil \right) \right]_{i=1}^{n} \tag{22}$$

- The "'Replicated CV" box support is taken as the average estimate over the B replicates:

$$\bar{\beta}^{rcv}(l) = \frac{1}{B} \sum_{b=1}^{B} \tilde{\beta}^{cv(b)}(l) \tag{23}$$

- Other "Replicated CV" box end-point quantities q estimates, taken as the average estimate over the B replicates:

$$\bar{q}^{rcv}(l) = \frac{1}{B} \sum_{b=1}^{B} \tilde{q}^{cv(b)}(l) \tag{24}$$

  This is done for: (i) The Minimal Event-Free Probability ($MEFP$): $\overline{P_0'}^{rcv}(l)$ and (ii) The Minimal Event-Free Time ($MEFT$): $\overline{T_0'}^{rcv}(l)$.

4 Numerical Analyses

4.1 Simulation Design

The p-dimensional covariates $\mathbf{x}_i = [x_{i,1} \dots x_{i,p}]^T$, for $i \in \{1, \dots, n\}$, were identically and independently drawn from either: a (i) p-multivariate normal distribution with mean vector $\boldsymbol{\mu}$ and variance-covariance matrix $\boldsymbol{\Sigma}$: $\mathbf{x}_i \sim N_p(\boldsymbol{\mu}, \boldsymbol{\Sigma})$; or (ii) from a p-multivariate uniform distribution on the interval $[a, b]$: $\mathbf{x}_i \sim U_p(a, b)$.

Simulations were carried out according to the assumptions stated in section 2.2.2. Simulated realizations of true survival times T_i were identically and independently drawn from an exponential distribution with rate parameter λ (and mean $\frac{1}{\lambda}$): $T_i \sim \text{Exp}(\lambda)$. Simulated realizations C_i of true censoring times were identically and independently sampled from a uniform distribution: $C_i \sim U(0, v)$ with $v > 0$, so that approximately $100 \times \pi(\%)$ of the simulated realizations of observed survival times $Y_i = \min(T_i, C_i)$ were censored, where $\pi \in \{0.3, 0.5, 0.7\}$. Finally, the simulated realizations of observed event (non-censoring) random variable indicator were $\delta_i = I(T_i \leqslant C_i)$.

To simulate survival models with various types of relationship between survival times (or hazards) and covariates (i.e. variable informativeness) including saturated regression models and noise models, the individual hazard rate parameter λ_i was simulated as an exponential regression function of individual covariate $\mathbf{x}_i$: $\lambda_i(t|\mathbf{x}_i) = \lambda_0(t) \exp[\eta(\mathbf{x}_i)]$, where the regression function $\eta(\mathbf{x}_i)$ can take different forms depending on the simulated survival model. In summary, our simulation was done using the following parameters (see documentation in our R package for more details [16]):

- $\mathbf{x}_i \sim U_p(0, 1)$ with $n = 250$ and $p = 3$, or $\mathbf{x}_i \sim N_p([0\ 0\ 0]^T, \sigma^2\mathbf{I})$ with $n = 100$ and $p = 1000$.
- by characterization of the first coverage box B_1 (i.e. for $m = 1$), using constrained/directed peeling, without pasting and with meta-parameter values $(\alpha_0, \beta_0) \in \{(0.10, 0.05)\}$.
- with censoring rate $\pi = 0.5$ and five concurrent simulated survival models, where $n > p$ and $n \ll p$, representing low- and high-dimensional situations, and where the regression function is as follows:
  - In simulated models #1-4, $\eta(\mathbf{x}_i) = \boldsymbol{\eta}^T\mathbf{x}_i$ with regression parameters $\boldsymbol{\eta} = [\eta_1 \dots 0_j \dots \eta_p]^T$, for $j \in \varnothing \cup \{1, \dots, p\}$:

$$\begin{cases} \text{Low-dim. Saturated Model \#1:} & n = 250, p = 3 & \boldsymbol{\eta} = [12\ \ -15\ \ -5]^T \\ \text{Low-dim. Un-saturated Model \#2:} & n = 250, p = 3 & \boldsymbol{\eta} = [12\ \ -15\ \ 0]^T \\ \text{Low-dim. Noise Model \#3:} & n = 250, p = 3 & \boldsymbol{\eta} = [0\ \ 0\ \ 0]^T \\ \text{High-dim. Un-saturated Model \#4:} & n = 100, p = 1000 & \boldsymbol{\eta} = [\eta_1 \dots \eta_{100}\ 0 \dots\ 0]^T \end{cases}$$

  - In simulated model #1b, a Low-dim. Saturated Model with $n = 250$ and $p = 3$, where

$$\eta(\mathbf{x}_i) = \begin{cases} \boldsymbol{\eta}^T\mathbf{x}_i \quad \text{with regression parameters } \boldsymbol{\eta} = [12\ \ -15\ \ -5]^T & \text{for} \quad \mathbf{x}_i \in R \\ u_i \sim U(0, 1) & \text{for} \quad \mathbf{x}_i \notin R \end{cases}$$

    where $R = [0.7, 1] \times [0, 0.2] \times [0, 0.4]$ is an arbitrary box in $\mathbb{R}^3$.
- using $K = 5$-fold cross-validation, $A = 1024$ for the permutation p-values and $B = 128$ independent replications.

4.2 Summary of Outputs

We explain below how the main diagnostic and descriptive output plots are used.

4.2.1 Cross-Validated Tuning Profiles

Cross-validated tuning profiles plot values of the box end-points statistics (section 2.2.6) Log Hazard Ratio (LHR), Log-Rank Test (LRT) or Concordance Error Rate (CER), depending on the optimization criterion chosen, as a function of peeling length or peeling steps of the peeling trajectory (model complexity). A peeling step includes step #0 corresponding to the situation where the starting box covers the entire test-set data $\mathcal{L}_k$ before peeling (Algorithm 1). These statistics are used internally or interactively to get the "Replicated CV" optimal length of the peeling trajectory: $\bar{L}^{rcv}$ (section 3.6). In order to successfully determine the profiles minimizer or maximizer (section 3.2.3), the cross-validated tuning profile should be approximately non-monotone up to sampling variability (Figure 4). In addition, one expects an inflation of variance of cross-validated point estimates towards the right-end of the cross-validated tuning profile corresponding to an increase in overfitting and model uncertainty for more complex models (Figure 4, Supporting_Figures 1, 2, 3).

The choice of the optimization criteria for controlling the peeling length is crucial. Two typical situations of failure of cross-validation can happen from the cross-validated tuning profiles of the box end-point statistics: either an extremum cannot be reached in the profile before the peeling sequence runs out of data, or the profile is essentially flat due to noise or the absence of any effect in the data (Figure 1). In the former case, this results in excessive peeling steps and cross-validated values of optimal peeling lengths $\bar{L}^{rcv}$ (eq. 19) at, or near, the maximal peeling lengths $\bar{L}^{rcv}_m$ (eq. 18). In the latter case, this results in un-reliable optimal peeling lengths $\bar{L}^{rcv}$ that can take any value between the $[1, \bar{L}^{rcv}_m]$ boundaries. In both cases, this leads to likely under-fitted or over-fitted models (see asterisk-annotated models in Table 1).

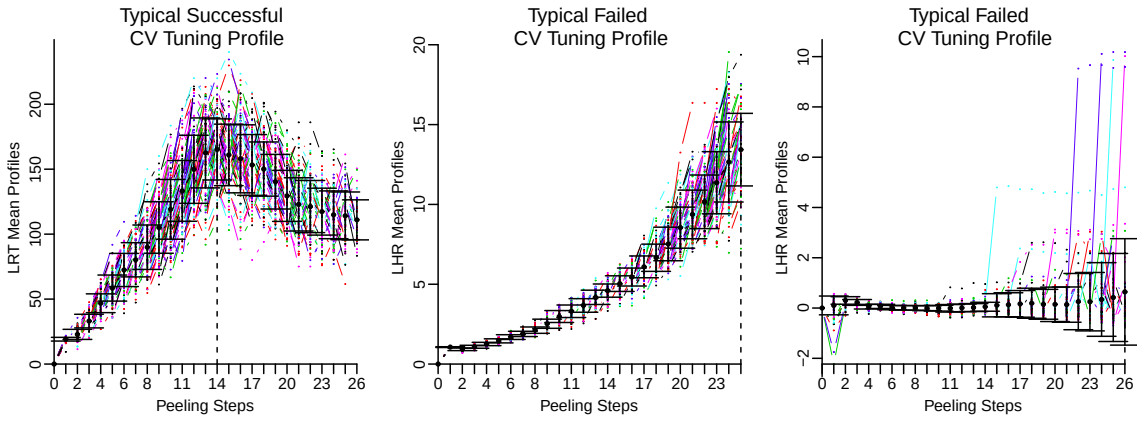


Figure 4: *Illustrations of typical successful (left) and failed (center and right) cross-validated tuning profiles of box end-point statistics. Left: Successful peeling stops with a "Replicated CV" optimal peeling length $\bar{L}^{rcv}$ (see eq. 19) reached within the $[1, \bar{L}^{rcv}_m]$ boundaries of possible peeling lengths; Center: Failure to reach a maximum before running out of data; Right: Failure to reach a reliable maximum because of a flat profile. The "Replicated CV" optimal peeling length $\bar{L}^{rcv}$ of the peeling trajectory is shown in each plot with the vertical black dashed line. Each colored profile corresponds to one of the replications ($B = 128$). The cross-validated mean profile of the statistic used in the optimization criterion is shown by the dotted black line with standard error of the sample mean.*

4.2.2 Peeling Trajectories

Cross-validated peeling trajectories are estimated by step functions of the covariates box cuts as a function of box support/mass (Figures 5, 7). They are read from right to left as they track the top-down peeling of the box induction process (peeling loop) of our "Patient Recursive Survival Peeling" method (Algorithm 1). These peeling trajectories are, up to sampling variability:

- Monotone (increasing or decreasing) functions for each input covariate $\mathbf{x}_j$, for $j \in \{1, \ldots, p\}$.
- Non-monotone (increasing then decreasing) functions for LHR $\bar{\lambda}^{rcv}(l)$.
- Non-monotone (increasing then decreasing) functions for LRT $\bar{\chi}^{rcv}(l)$.
- Non-monotone (decreasing then increasing) functions of CER $\bar{\theta}^{rcv}(l)$.

- Monotone decreasing functions for *MEFP* $\overline{P}_0'^{rcv}(l)$.
- Monotone decreasing functions for *MEFT* $\overline{T}_0'^{rcv}(l)$.

4.2.3 Trace Curves

Cross-validated trace curves of covariate importance and covariate usage are estimated by piece-wise linear and step functions, respectively, as a function of box support/mass (Figures 6, 8). Similarly to peeling trajectories, they are read from right to left. Trace curves of covariate importance show on a single plot: (i) the amplitude of used covariates, (ii) the order (prioritization) with which these covariates are used, and (iii) the extent of the number of peeling steps by which each covariate is used. Covariate traces are reminiscent of the concept of variable selection from the fields of decision tree and regularization, that is:

- In "Variable Importance", a prediction-based statistics borrowed from the existing theory of decision trees [9] and their ensemble version [8].
- In "Selective Shrinkage" of variable coefficients/parameters from the existing theory of regularization and variable selection (e.g. LARS [25], Lasso [70], Elastic Net [74] and Spike & Slab [41]).

4.2.4 Survival Curves

Each subplot of Figure 9, 11 and 13 corresponds to a peeling step of our Patient Recursive Survival Peeling method for a tested model and cross-validation technique (including none). Each subplot shows cross-validated Kaplan–Meir estimates of the survival functions, as a function of survival time, of both "in-box" (red) and "out-of-box" (black) samples, corresponding respectively to the high-risk and low-risk groups. Each subplot also displays the corresponding step number along with cross-validated Log Hazard Ratio (*LHR*), Log-Rank Test (*LRT*) and log-rank permutation p-value $\tilde{p}^{cv}(l)$ of survival distribution separation (see section 3.5). A single survival curve always exists at Step #0 corresponding to the situation where the starting box covers the entire test-set data $\mathcal{L}_k$ before peeling (Algorithm 1). As the peeling progresses, the survival curves of "in-box" and "out-of-box" samples further separate until the peeling stops.

4.3 Effect of Model Tuning

4.3.1 Effect of the Optimization Criteria

We first compare the effect of the three optimization criteria (Log Hazard Ratio *LHR*, Log-Rank Test *LRT* or Concordance Error Rate *CER* - section 3.2.3) used for model tuning and selection. Evaluations are reported on (i) the "Replicated CV" optimal peeling lengths $\bar{L}^{rcv}$ (eq. 19) obtained from the cross-validated tuning profiles of the box end-point statistics (Log Hazard Ratio *LHR*, Log-Rank Test *LRT* or Concordance Error Rate *CER*), and (ii) on the cross-validated numbers of used covariates by the PRSP algorithm (see Algorithm 1) out of the total number of pre-selected ones. Results are presented in Table 1, Supporting_Table 1 and Supporting_Figures 1, 2, 3 for the three peeling criteria used (Log Hazard Ratio *LHR*, Log-Rank Test *LRT* or Cumulative Hazard Summary *CHS* - section 2.2.3) as well as for our two cross-validation techniques ("Replicated Averaged CV" (RACV) vs. "Replicated Combined CV" (RCCV)), whether in low- or high-dimensional simulated survival regression models #1, #2, #3 and #4.

From Table 1 and Supporting_Figures 1, 2, 3, it results that both Log-Rank Test (*LRT*) and Concordance Error Rate (*CER*) optimization criteria give satisfactory results in low-dimensional simulated models (#1 and #2), other than the simulated noise model (#3), where the peeling length is optimally pruned ($\bar{L}^{rcv} = 9 - 20$). This is in sharp contrast to the Log Hazard Ratio (*LHR*) optimization criterion that fails to control the peeling length, regardless of the cross-validation technique or the peeling criterion used ($\bar{L}^{rcv} = 26 - 27$).

The situation differs in high-dimensional simulated models: the Concordance Error Rate (*CER*) appears to be the only optimization criterion that reliably controls the peeling length in simulated model #4, regardless of the peeling criterion or cross-validation technique used (Table 1, Supporting_Figures 1, 2, 3).

Finally, note that the Concordance Error Rate *CER*) optimization criterion tends to yield slightly more conservative results than the Log-Rank Test (*LRT*) in terms of peeling length (Table 1, Supporting_Figures

Table 1: *Effect of peeling and optimization criteria as well as cross-validation techniques on the cross-validated tuning profiles of the box end-point statistics (Log Hazard Ratio LHR, Log-Rank Test LRT or Concordance Error Rate CER) and the resulting "Replicated CV" optimal peeling length $\bar{L}^{rcv}$ (see eq. 19). Cross-validated optimal peeling lengths $\bar{L}^{rcv}$ are reported for the combined effects of: (i) the three peeling criteria (by rows: Log Hazard Ratio (LHR), Log-Rank Test (LRT) or Cumulative Hazard Summary (CHS)), (ii) the three optimization criteria (by columns: Log Hazard Ratio (LHR), Log-Rank Test (LRT) or Concordance Error Rate (CER)), (iii) the two cross-validation techniques (by columns: "Replicated Averaged CV" or RACV and "Replicated Combined CV" or RCCV), and (iv) the four tested simulation models (by rows: Model #1, #2, #3 or #4). Asterisks denote situations where $\bar{L}^{rcv}$ reaches either a (quasi-)minimal or (quasi-)maximal value of optimal peeling lengths, corresponding to failed cross-validations / likely under-fitted or over-fitted models.*

Model #1

		Optimization Criterion					
		LHR		*LRT*		*CER*	
		RACV	RCCV	RACV	RCCV	RACV	RCCV
Peeling Criterion	*LHR*	26*	25*	20	20	10	11
	LRT	26*	25*	17	20	10	10
	CHS	26*	26*	14	14	09	09

Model #2

		Optimization Criterion					
		LHR		*LRT*		*CER*	
		RACV	RCCV	RACV	RCCV	RACV	RCCV
Peeling Criterion	*LHR*	26*	25*	17	17	12	12
	LRT	26*	24*	10	11	10	10
	CHS	26*	26*	14	14	10	10

Model #3

		Optimization Criterion					
		LHR		*LRT*		*CER*	
		RACV	RCCV	RACV	RCCV	RACV	RCCV
Peeling Criterion	*LHR*	23*	01	23*	01	05	02
	LRT	26*	02	26*	02	03	02
	CHS	24*	01	23*	02	04	02

Model #4

		Optimization Criterion					
		LHR		*LRT*		*CER*	
		RACV	RCCV	RACV	RCCV	RACV	RCCV
Peeling Criterion	*LHR*	17*	09	16*	06*	05	05
	LRT	16*	01*	16*	08*	06	08
	CHS	16*	01*	16*	08*	05	05

1, 2, 3) and number of used covariates (Supporting_Table 1). Also, note that the Concordance Error Rate *CER*) has systematically less variance than the other Log-Rank Test (*LRT*) and Log Hazard Ratio (*LHR*) optimization criteria (Table 1, Supporting_Figures 1, 2, 3).

For the above reasons, we recommend using the Concordance Error Rate *CER* as optimization criterion in every situation or the Log-Rank Test *LRT* in low-dimensional situation only.

4.3.2 Effect of the Peeling Criteria

Overall, in all simulation models tested, the effect of the peeling criteria, used for model fitting of the survival bump hunting model, is relatively marginal compared to that of the optimization criterion and/or cross-validation technique used for model tuning.

However, for any combination of the optimization criterion and cross-validation technique used, both Log-Rank Test *LRT* and Cumulative Hazard Summary *CHS* peeling criteria tend to induce slightly shorter profiles and use less covariates than the Log Hazard Ratio *LHR* criterion (Table 1, Supporting_Table 1,

Supporting_Figures 1, 2, 3). Moreover, the Cumulative Hazard Summary *CHS* peeling criterion tends to be slightly more conservative than the Log-Rank Test *LRT* (and the Log Hazard Ratio *LHR*) in terms of peeling length and number of used covariates, especially in high-dimensional models (Table 1, Supporting_Table 1, Supporting_Figures 1, 2, 3).

So, we recommend using primarily the Cumulative Hazard Summary *CHS* and secondly the Log-Rank Test *LRT* as peeling criterion (in low or high-dimensional data) to reduce the risk of over-fitting (or conversely the Log Hazard Ratio *LHR* to avoid excessive conservativeness).

4.3.3 Effect of the Cross-Validation Technique

The difference of results between cross-validation techniques (including none) is striking in simulated noise model #3. Here, only the "Replicated Combined CV" (RCCV) cross-validation technique gives satisfactory results, regardless of the optimization or peeling criterion used. As expected in this situation, the model is extensively pruned ($\bar{L}^{rcv} = 1 - 2$) using RCCV. The same consistency is not observed for "Replicated Averaged CV" (RACV) that fails to properly control the peeling length when the other Log-Rank Test (*LRT*) or Log Hazard Ratio (*LHR*) optimization criteria are used ($\bar{L}^{rcv} \approx \bar{L}^{rcv}_m$) (Table 1, Supporting_Figures 1, 2, 3). Differences between cross-validation techniques are also significant in high-dimensional simulated models (Table 1, Supporting_Figures 1, 2, 3).

Using from now on a given peeling and optimization criterion in low-dimensional simulated models, we compare the performance of our two cross-validation techniques with each other (RACV vs. RCCV) and with the situation of no cross-validation (NOCV). Peeling trajectory and covariate usage/importance results are shown for model #2 where it is possible to specifically assess effects of cross-validation on a noise/random covariate ($\mathbf{x}_3$). Figures 5 and 6 show peeling trajectory profiles and covariate traces for model #2. Table 2 gives the corresponding rules.

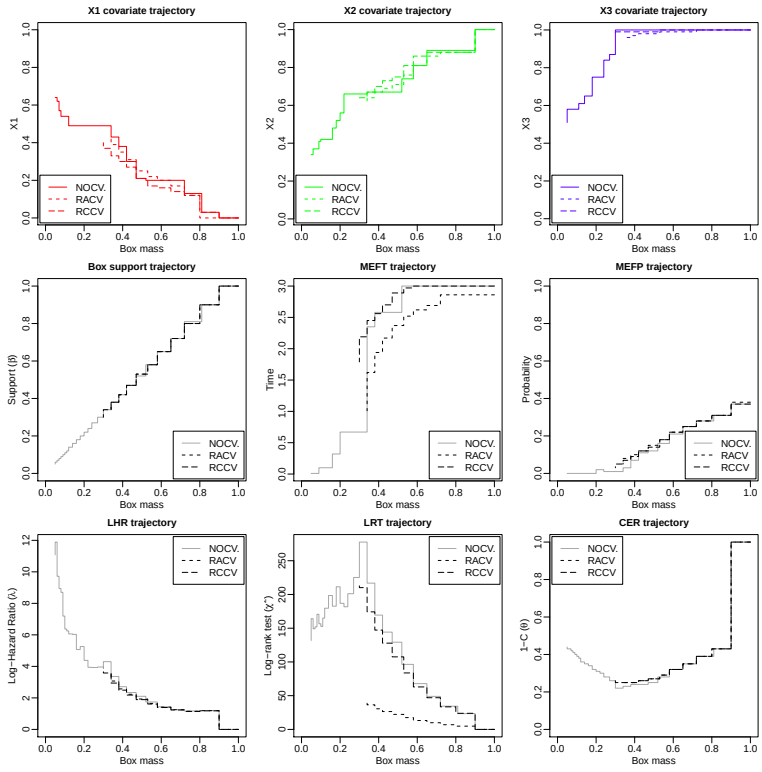


Figure 5: *Comparison of cross-validated peeling trajectories between situations when either cross-validation technique "Replicated Combined CV" (RCCV) or "Replicated Averaged CV" (RACV) and no cross-validation (NOCV) was done. Results are for simulated model #2 and the LRT statistic used in both peeling and optimization criteria. Compare the trajectory lengths between either cross-validation technique and in the absence of either one. Notice also the flat trajectory profile of covariate* $\mathbf{x}_3$ *in the presence of either cross-validation technique (RACV or RCCV) as opposed to the situation where no cross-validation (NOCV) was done.*

Clearly, both cross-validation techniques are effective in terms of (i) smoothing peeling trajectories out and (ii) pruning peeling trajectories off. Compare for instance results of simulation model #2: $\bar{L}^{rcv} = 26$ without cross-validation (NOCV), $\bar{L}^{rcv} = 10$ with RACV and $\bar{L}^{rcv} = 11$ with RCCV (Figures 5 and 6, Table 2). In fact, all cross-validated trajectory profiles in Figure 5 and covariate traces in Figure 6 stop at $\bar{\beta}^{rcv}(l = 11) \lessapprox 0.30$ for RCCV and $\bar{\beta}^{rcv}(l = 10) \lessapprox 0.34$ for RACV as compared to $\bar{\beta}^{rcv}(l = 27) \lessapprox 0.05$ in the absence of cross-validation (NOCV).

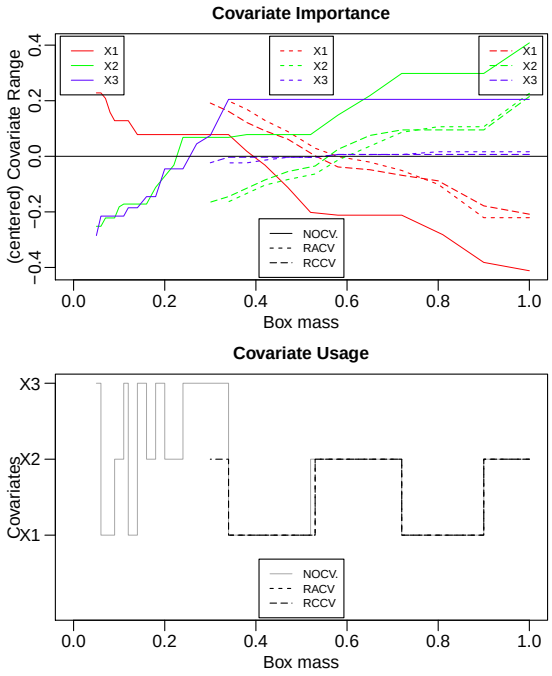


Figure 6: *Comparison of cross-validated trace plots of covariate importance $\overline{VI}(l)$ (top) and covariate usage $\overline{VU}(l)$ (bottom) between situations when either cross-validation technique "Replicated Combined CV" (RCCV) or "Replicated Averaged CV" (RACV) and no cross-validation (NOCV) was done. Results are for simulated model #2 and the LRT statistic used in both peeling and optimization criteria. Compare the trace lengths between either cross-validation technique and in the absence of either one. Notice also the flat trace of covariate $\mathbf{x}_3$ about 0 in the presence of either cross-validation technique (RACV or RCCV) as opposed to the situation where no cross-validation (NOCV) was done.*

Notice from the peeling trajectories and trace plots of simulation model #2 (compared to #1) how a noise/random covariate ($\mathbf{x}_3$) is effectively eliminated from the model after using either cross-validation technique (RCCV or RACV), while it is not in the absence of cross-validation (NOCV). In fact, $\mathbf{x}_3$'s RCCV and RACV peeling trajectories are mostly flat, that is, $\mathbf{x}_3$ is unused in the decision rule (blue dashed curves in Figure 5). Consistently, $\mathbf{x}_3$'s RCCV and RACV covariate importance trace plots are mostly flat (top of Figure 6) and $\mathbf{x}_3$'s RCCV and RACV covariate usage trace plots show that $\mathbf{x}_3$ is not used at all (bottom of Figure 6). Similar conclusions are drawn with respect to simulated model #3 (compared to #1).

The non-monotone behavior of the *LRT* and the overly large *LHR* values obtained in the non-cross-validated (NOCV) results of simulated model #2 ($\bar{\lambda}^{rcv}(l = 26) = 11.08$) clearly reflect over-fitting and sub-optimal models. This is evident when comparing to the much more conservative values obtained from the corresponding cross-validated peeling profiles: $\bar{\lambda}^{rcv}(l = 11) = 3.90$ for RCCV and $\bar{\lambda}^{rcv}(l = 10) = 3.76$ for RACV (Figure 5 and Table 2). This non-monotone behavior of *LRT* peeling profile is precisely what allows us to use it in the optimization criterion. We suggest that this could be due to a greater sensitivity of *LRT* to small sample sizes at deep peeling steps.

To further compare cross-validation techniques, we generated empirical distributions of various cross-validated estimates of box decision rules and box survival end-points/prediction statistics (section 2.2.6) for each technique and end-point as a function of peeling steps. Distributions were obtained by generating $B = 128$ Monte-Carlo simulated datasets according to simulated model #1, where the *LRT* statistic was

Table 2: *Comparison of cross-validated decision rules (upper Table) and box end points statistics of interest (lower Table) between situations when either cross-validation technique "Replicated Combined CV" (RCCV) or "Replicated Averaged CV" (RACV) and no cross-validation (NOCV) was done. For conciseness, only the initial and final decision rules ($\bar{L}^{rcv}$th step) are shown. Values are sample mean estimates with corresponding standard errors in parenthesis (NA in the case of NOCV, where no replication was performed - see manual of R package* `PRIMsrc` *for details [16]). Step #0 corresponds to the situation where the starting box covers the entire test-set data $\mathcal{L}_k$ before peeling. Results are for simulated model #2 and the LRT statistic used in both peeling and optimization criteria. Notice the non-usage of covariate $\mathbf{x}_3$ in the presence of either cross-validation technique (RACV or RCCV) as opposed to the situation where no cross-validation (NOCV) was done.*

	Step l	$\mathbf{x}_1$	$\mathbf{x}_2$	$\mathbf{x}_3$
RCCV	0	$\mathbf{x}_1 \geqslant 0.00$ (0.00)	$\mathbf{x}_2 \leqslant 1.00$ (0.00)	$\mathbf{x}_3 \leqslant 1.00$ (0.00)
	1	$\mathbf{x}_1 \geqslant 0.03$ (0.00)	$\mathbf{x}_2 \leqslant 0.88$ (0.01)	$\mathbf{x}_3 \leqslant 1.00$ (0.00)
	⋮	⋮	⋮	⋮
	11	$\mathbf{x}_1 \geqslant 0.40$ (0.06)	$\mathbf{x}_2 \leqslant 0.62$ (0.04)	$\mathbf{x}_3 \leqslant 0.97$ (0.05)
RACV	0	$\mathbf{x}_1 \geqslant 0.00$ (0.00)	$\mathbf{x}_2 \leqslant 1.00$ (0.00)	$\mathbf{x}_3 \leqslant 1.00$ (0.00)
	1	$\mathbf{x}_1 \geqslant 0.00$ (0.00)	$\mathbf{x}_2 \leqslant 0.88$ (0.00)	$\mathbf{x}_3 \leqslant 1.00$ (0.00)
	⋮	⋮	⋮	⋮
	10	$\mathbf{x}_1 \geqslant 0.42$ (0.03)	$\mathbf{x}_2 \leqslant 0.61$ (0.02)	$\mathbf{x}_3 \leqslant 0.96$ (0.03)
NOCV	0	$\mathbf{x}_1 \geqslant 0.00$ (NA)	$\mathbf{x}_2 \leqslant 1.00$ (NA)	$\mathbf{x}_3 \leqslant 1.00$ (NA)
	1	$\mathbf{x}_1 \geqslant 0.03$ (NA)	$\mathbf{x}_2 \leqslant 0.89$ (NA)	$\mathbf{x}_3 \leqslant 1.00$ (NA)
	⋮	⋮	⋮	⋮
	26	$\mathbf{x}_1 \geqslant 0.64$ (NA)	$\mathbf{x}_2 \leqslant 0.34$ (NA)	$\mathbf{x}_3 \leqslant 0.51$ (NA)

	Step l	$n(l)$	$\bar{\beta}^{rcv}(l)$	$T_0'^{rcv}(l)$	$P_0'^{rcv}(l)$	$\bar{\lambda}^{rcv}(l)$	$\bar{\chi}^{rcv}(l)$	$\bar{\theta}^{rcv}(l)$
RCCV	0	250 (0.00)	1.00 (0.00)	3.00 (0.00)	0.37 (0.00)	0.00 (0.00)	0.00 (0.00)	1.00 (0.00)
	1	225 (0.00)	0.90 (0.00)	3.00 (0.00)	0.31 (0.01)	0.18 (0.02)	23.86 (1.14)	0.43 (0.00)
	⋮	⋮	⋮	⋮	⋮	⋮	⋮	⋮
	11	75 (2.50)	0.30 (0.01)	1.79 (0.71)	0.03 (0.02)	3.90 (0.46)	214.61 (33.56)	0.26 (0.01)
RACV	0	250 (0.00)	1.00 (0.00)	2.86 (0.04)	0.38 (0.02)	0.00 (0.00)	0.00 (0.00)	1.00 (0.00)
	1	225 (2.50)	0.90 (0.01)	2.86 (0.05)	0.31 (0.02)	1.19 (0.02)	4.84 (0.27)	0.43 (0.01)
	⋮	⋮	⋮	⋮	⋮	⋮	⋮	⋮
	10	85 (2.50)	0.34 (0.01)	1.01 (0.34)	0.05 (0.02)	3.76 (0.41)	43.64 (6.01)	0.25 (0.02)
NOCV	0	250 (NA)	1.00 (NA)	3.00 (NA)	0.37 (NA)	0.00 (NA)	0.00 (NA)	1.00 (NA)
	1	225 (NA)	0.90 (NA)	3.00 (NA)	0.31 (NA)	1.19 (NA)	23.43 (NA)	0.43 (NA)
	⋮	⋮	⋮	⋮	⋮	⋮	⋮	⋮
	26	12 (NA)	0.05 (NA)	0.01 (NA)	0.00 (NA)	11.08 (NA)	131.73 (NA)	0.44 (NA)

used in both peeling and optimization criteria. The replication design accounts for two folds of variability: the one due to random splitting by cross-validation and the one due to sampling from the simulated model. Then, Box Coefficient of Variation (BCV) of decision rules (as defined in [17]) and coefficient of variation of box survival end-points/prediction statistics were computed and plotted as a function of peeling steps. Here, cross-validated coefficient of variation profiles do not show a consistent advantage of one cross-validation technique over the other considering all end-point analyzed. This remains true for a range of realistic sample sizes $n \in \{50, 100, 200\}$ (Supporting_Figure 4).

Overall, our two cross-validation techniques, although not equivalent in design, give similar results on most profiles for the sample size and simulation models tested, confirming that both techniques are appropriate to the task in most situations. However, "Replicated Averaged CV" (RACV) appears to be less conservative than "Replicated Combined CV" (RCCV), especially in high-dimensional settings, which could be a problem if ones cares about reducing the risk of over-fitting. Also, RACV failed to prune simulated model #3 (Table 1 and Supporting_Figures 1, 2, 3). This indicates that our RCCV cross-validation technique is more robust in noisy situations, possibly because RCCV uses larger test-set samples of size n to make estimations than RACV, which uses test-set samples of size $n^t \approx n/K$ (section 3.2.2). For these reasons, we recommend using RCCV preferably to RACV.

4.3.4 Comparison Between Simulated Survival Models

In line with the above guidelines (sections 4.3.2 and 4.3.3), we used the following criteria and techniques in our numerical analyses for fitting and tuning/selecting our survival bump hunting model: (i) The Log-Rank Test *LRT* both as peeling and optimization criterion in low-dimensional settings; (ii) The Cumulative Hazard Summary *CHS* as peeling criterion and the Concordance Error Rate *CER* as optimization criteria in high-dimensional settings; and our "Replicated Combined Cross-Validation" (RCCV) technique. We compared the performance of our Survival Bump Hunting procedure in terms of peeling trajectories (Figures 7, 8, Table 3, Supporting_Figures 5, 6, Supporting_Table 2) and survival distribution curves (Figure 9) between all our models.

Notice the striking differences in cross-validated peeling trajectories (Figure 7) and covariate traces (Figure 8): (i) when all covariates $(\mathbf{x}_1, \mathbf{x}_2, \mathbf{x}_3)$ are noise (model #3), or (ii) when one covariate only $(\mathbf{x}_3)$ is noise (model #2) or (iii) when all are informative (model #1). As expected, *all* peeling trajectories related to model #3 are much shorter than in the other models, indicating an abortive PRSP procedure with little or no covariate usage during the peeling process (Figure 7) nor involvement in the decision rule (Table 3). Similarly, one expects little or no usage of covariate $\mathbf{x}_3$ in models #2 and #3, as seen in their covariate peeling trajectories (Figure 7, Table 3).

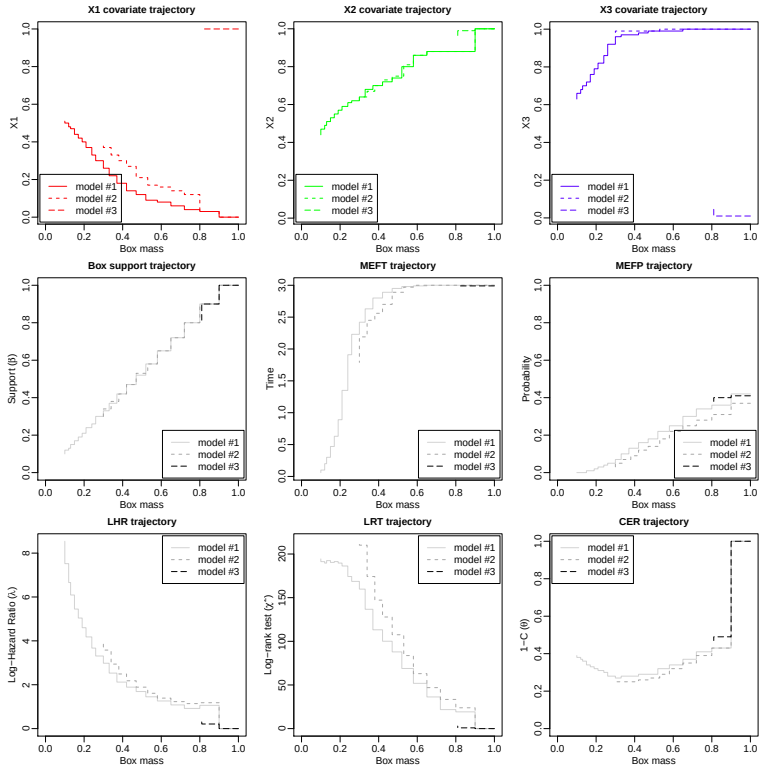


Figure 7: *Comparison of replicated combined cross-validated results for the peeling trajectories between simulated models #1, #2 and #3 for the "Replicated Combined CV" (RCCV) technique and the LRT statistic used in both peeling and optimization criteria. Notice the usage of all covariates* $(\mathbf{x}_1, \mathbf{x}_2, \mathbf{x}_3)$ *in model #1 as opposed to the selective usage of covariates* $(\mathbf{x}_1, \mathbf{x}_2)$ *in model #2 and the abortive usage of all covariates in noise model #3.*

Consistent observations can be made from the cross-validated covariate importance and covariate usage traces of model #3. In fact, $\mathbf{x}_3$'s covariate importance trace stops after only $\bar{L}^{rcv} = 2$ steps with box mass $\bar{\beta}^{rcv}(\bar{L}^{rcv}) \approx 0.81$ for model #3, as compared to $\bar{L}^{rcv} = 20$ steps with $\bar{\beta}^{rcv}(\bar{L}^{rcv}) \approx 0.10$ for model #1 and $\bar{L}^{rcv} = 11$ steps with $\bar{\beta}^{rcv}(\bar{L}^{rcv}) \approx 0.30$ for model #2 (Table 3 and Figure 8). Also notice the limited usage of covariate $\mathbf{x}_3$ in the covariate usage traces of models #2 and #3 as compared to #1 and the fact that all cross-validated peeling trajectories in model #1 extend further than in other models #2 and #3 (Figure 7, 8 and Table 3). This is consistent with our simulation design in that all covariates in regression model

#1 additively contribute to the hazards and to the separation of survival distributions.

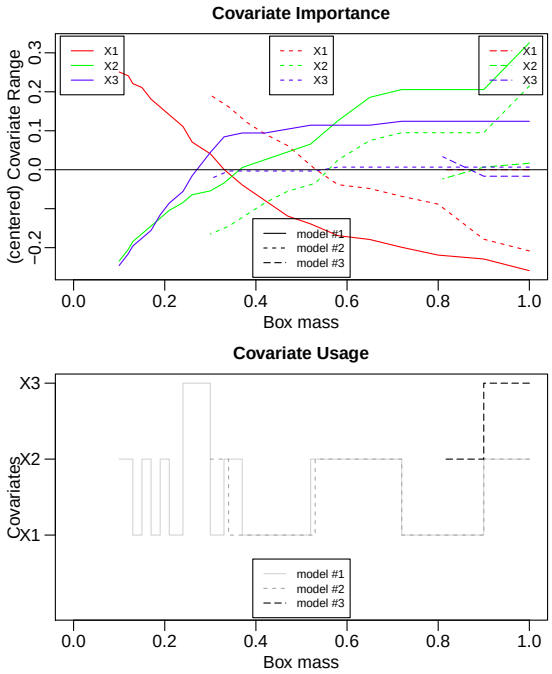


Figure 8: *Comparison of replicated combined cross-validated trace plots of covariate importance* $\overline{VI}(l)$ *(top) and covariate usage* $\overline{VU}(l)$ *(bottom) between simulated models #1, #2 and #3 for the "Replicated Combined CV" (RCCV) technique and the LRT statistic used in both peeling and optimization criteria. Notice the usage of all covariates* $(\mathbf{x}_1, \mathbf{x}_2, \mathbf{x}_3)$ *in model #1 as opposed to the selective usage of covariates* $(\mathbf{x}_1, \mathbf{x}_2)$ *in model #2 and the abortive usage of all covariates in noise model #3.*

Finally, Figure 9 shows the cross-validated Kaplan–Meir survival probability curves of the highest-risk group vs. lower-risk group in all low- and high-dimensional simulated models with their corresponding log-rang permutation p-values of survival distribution curve separation. The separation is especially evident in results of models #1, #2 and #4 in contrast to the overlap seen in model #3. The permutation p-values are: $\tilde{p}^{cv}(l = 20) \leqslant 9.7e-5$, $\tilde{p}^{cv}(l = 11) \leqslant 9.7e-5$ and $\tilde{p}^{cv}(l = 8) \approx 0.046$ for models #1, #2 and #4 respectively, and $\tilde{p}^{cv}(l = 2) \approx 0.1080$ for model #3 (Figure 9).

Overall, Figures 7, 8, 9, Table 3, Supporting_Figures 5, 6 and Supporting_Table 2 collectively support that our "Replicated Combined CV" (RCCV) cross-validation technique, when used with an appropriate combination of *LRT* and *CER* statistics as peeling and/or optimization criteria, is efficient at fitting and tuning a survival bump hunting model, whether in low- or high-dimensional settings. Moreover, we show that our PRSP algorithm (Algorithm 1) successfully selects/uses a subset of (or all) the covariates that are informative (i.e. that truly enter into the model) in the box decision rules.

Table 3: *Comparison of cross-validated decision rules (upper Table) and box end points statistics of interest (lower Table) between simulated models #1, #2 and #3 for the "Replicated Combined CV" (RCCV) technique and the LRT statistic used in both peeling and optimization criteria. For conciseness, only the initial and final decision rules ($\bar{L}^{rcv}$th step) are shown. Step #0 corresponds to the situation where the starting box covers the entire test-set data $\mathcal{L}_k$ before peeling. Values are sample mean estimates with corresponding standard errors in parenthesis. Notice the usage of all covariates $(\mathbf{x}_1, \mathbf{x}_2, \mathbf{x}_3)$ in model #1 as opposed to the selective usage of covariates $(\mathbf{x}_1, \mathbf{x}_2)$ in model #2 and the abortive usage of all covariates in noise model #3.*

	Step l	$\mathbf{x}_1$	$\mathbf{x}_2$	$\mathbf{x}_3$
model #1	0	$\mathbf{x}_1 \geqslant 0.00$ (0.00)	$\mathbf{x}_2 \leqslant 1.00$ (0.00)	$\mathbf{x}_3 \leqslant 1.00$ (0.00)
	1	$\mathbf{x}_1 \geqslant 0.03$ (0.00)	$\mathbf{x}_2 \leqslant 0.88$ (0.00)	$\mathbf{x}_3 \leqslant 1.00$ (0.00)
	⋮	⋮	⋮	⋮
	20	$\mathbf{x}_1 \geqslant 0.51$ (0.04)	$\mathbf{x}_2 \leqslant 0.44$ (0.06)	$\mathbf{x}_3 \leqslant 0.63$ (0.08)
model #2	0	$\mathbf{x}_1 \geqslant 0.00$ (0.00)	$\mathbf{x}_2 \leqslant 1.00$ (0.00)	$\mathbf{x}_3 \leqslant 1.00$ (0.00)
	1	$\mathbf{x}_1 \geqslant 0.03$ (0.00)	$\mathbf{x}_2 \leqslant 0.88$ (0.00)	$\mathbf{x}_3 \leqslant 1.00$ (0.00)
	⋮	⋮	⋮	⋮
	11	$\mathbf{x}_1 \geqslant 0.40$ (0.06)	$\mathbf{x}_2 \leqslant 0.62$ (0.04)	$\mathbf{x}_3 \leqslant 0.97$ (0.05)
model #3	0	$\mathbf{x}_1 \leqslant 1.00$ (0.00)	$\mathbf{x}_2 \leqslant 1.00$ (0.00)	$\mathbf{x}_3 \geqslant 0.01$ (0.00)
	1	$\mathbf{x}_1 \leqslant 1.00$ (0.00)	$\mathbf{x}_2 \leqslant 0.99$ (0.01)	$\mathbf{x}_3 \geqslant 0.01$ (0.01)
	2	$\mathbf{x}_1 \leqslant 1.00$ (0.00)	$\mathbf{x}_2 \leqslant 0.96$ (0.05)	$\mathbf{x}_3 \geqslant 0.06$ (0.02)

	Step l	$n(l)$	$\bar{\beta}^{rcv}(l)$	$T_0'^{rcv}(l)$	$P_0'^{rcv}(l)$	$\bar{\lambda}^{rcv}(l)$	$\bar{\chi}^{rcv}(l)$	$\bar{\theta}^{rcv}(l)$
model #1	0	250 (0.00)	1.00 (0.00)	3.00 (0.00)	0.42 (0.00)	0.00 (0.00)	0.00 (0.00)	1.00 (0.00)
	1	225 (0.00)	0.90 (0.00)	3.00 (0.00)	0.36 (0.00)	1.06 (0.03)	19.21 (1.13)	0.43 (0.00)
	⋮	⋮	⋮	⋮	⋮	⋮	⋮	⋮
	20	25 (2.50)	0.10 (0.01)	0.06 (0.05)	0.00 (0.00)	8.54 (1.29)	194.39 (30.24)	0.39 (0.01)
model #2	0	250 (0.00)	1.00 (0.00)	3.00 (0.00)	0.37 (0.00)	0.00 (0.00)	0.00 (0.00)	1.00 (0.00)
	1	225 (0.00)	0.90 (0.00)	3.00 (0.00)	0.31 (0.00)	1.18 (0.02)	23.86 (1.14)	0.43 (0.00)
	⋮	⋮	⋮	⋮	⋮	⋮	⋮	⋮
	11	75 (2.50)	0.30 (0.01)	1.79 (0.71)	0.03 (0.71)	3.90 (0.46)	214.61 (33.56)	0.26 (0.01)
model #3	0	250 (0.00)	1.00 (0.00)	2.99 (0.00)	0.41 (0.00)	0.00 (0.00)	0.00 (0.00)	1.00 (0.00)
	1	225 (2.50)	0.90 (0.01)	2.99 (0.00)	0.40 (0.01)	0.21 (0.14)	0.86 (0.79)	0.49 (0.01)
	2	202 (5.00)	0.81 (0.02)	2.99 (0.00)	0.38 (0.01)	0.33 (0.11)	2.90 (1.75)	0.47 (0.01)

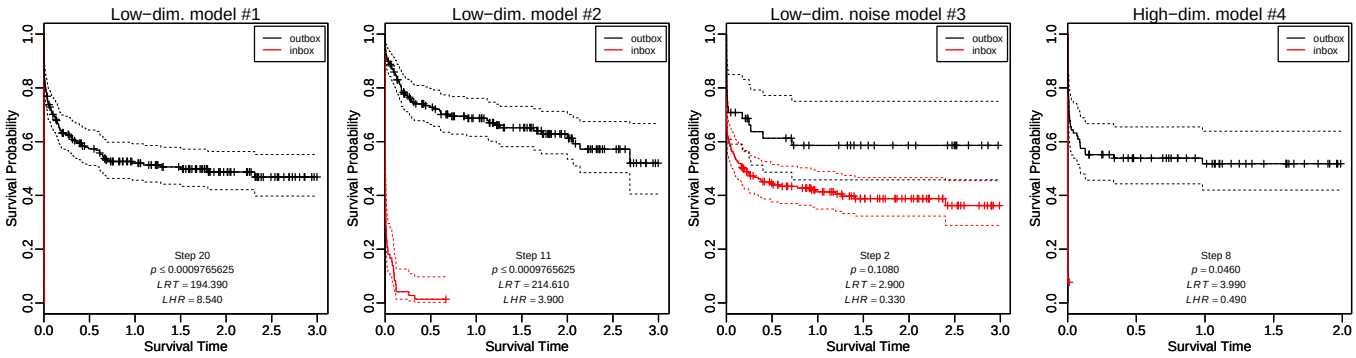


Figure 9: *Comparison of cross-validated Kaplan–Meir survival probability curves of the high-risk (red curve "in-box") and low-risk (black curve "out-of-box") groups in simulated models #1, #2, #3 and #4. Results are for the "Replicated Combined CV" (RCCV) technique and the CHS statistic used as peeling criterion and CER used as optimization criteria. Left column: model #1, middle column: model #2, right column: model #3. For conciseness, only the last peeling step of the peeling sequence is shown for each model. Cross-validated LRT, LHR and permutation p-values of "in-box" samples are shown at the bottom of the plot with the corresponding peeling step for each method. P-values $\hat{p}^{cv}(l) \leqslant 9.7e-5$ correspond to 1/10th of the precision limit (see section 3.5). Notice how the survival curves of "in-box" and "out-of-box" samples separates in models #1, #2 and #4 in contrast to the overlapping situation in noise model #3 with the corresponding significant and non-significant log-rank permutation p-value $\hat{p}^{cv}(l)$ of survival distribution separation.*

4.4 Comparisons Against Other Methods

4.4.1 Design and Choice of Various Non-Parametric Survival Models

Next, we compared our Survival Bump Hunting (SBH) procedure by our PRSP algorithm (1) to other competitive non-parametric survival models or methods in terms of survival and prediction end-points statistics. In all our performance analyses below, we used LRT in both peeling and optimization criteria and RCCV as our cross-validation technique. Comparisons include (i) Survival Bump Hunting by our PRSP method (Algorithm 1 and [14]), (ii) Regression Survival Trees (RST) by recursive partitioning [1, 11, 13, 32, 45, 46, 62], (iii) Random Survival Forest (RSF) by ensemble tree-based method [40], (iv) Cox Proportional Hazard Regression (CPHR) [12], (v) Survival Supervised PCA (SSPCA) [4], (vi) Survival Supervised Clustering (SSC) [5].

The simulated survival model was according to simulated model #1b, that is, by generating a box-shaped region R of the input covariate space with higher hazards than a uniform background (see 4.1). For comparisons, $B = 128$ repeated Monte-Carlo simulated datasets #1b were used to generate empirical sampling distributions of cross-validation estimates of box statistics, survival end-points and prediction performance metrics (section 2.2.6) and make points and confidence intervals inferences. This replication design accounts for random splitting and simulated model sampling variabilities.

For each method, an internal cross-validation was carried out for model fitting/training that was done by optimizing a specific empirical objective function of Goodness of Fit or Prediction Error measure on the corresponding test-set, such as: (i) maximization of the Log-Rank Test statistic or minimization of a Concordance Error Rate (SBH), maximization of the Deviance Residuals statistic (RST), maximization of the Concordance Index (RSF), maximization of the Likelihood Ratio Statistic between the reduced vs. full model (SSPCA), maximization of the Concordance Index (SSC). Then, cross-validation was used again to make estimations and predictions on the combined test-sets as described before (section 3.2.2).

Whether the goal is to make estimations or predictions, one wants to classify samples into two survival/risk groups. However, unlike Survival Bump Hunting that inherently generates "in-box" and "out-of-box" groups, all other methods do not necessarily give directly two survival/risk groups. For comparisons purposes, one needs to come up with a calibrated way across all other methods to output two groups only. One way, shown to work well empirically, is by using the median survival time threshold ([38]).

4.4.2 Comparison of End-Point Estimates

Specifically, the trained models generate cross-validated fits from which cross-validated estimates of highest-risk/group support and survival end-points statistics (described in section 2.2.6) are made using the left-out test-set $\mathcal{L}_k$. We report the results for all methods in Figure 10 below. The figure shows the highest-risk/group end-points distributions of RCCV estimates of support and survival end-points statistics computed over $B = 128$ repeated Monte-Carlo simulated models #1b for all competitive non-parametric survival models under study.

In Figure 11 below, a Kaplan–Meier estimate of RCCV survival probability curve is shown for each group and competitive non-parametric survival models under study from one replicate out of $B = 128$. As expected, it shows the extremeness of the survival distribution of the highest-risk box/group found by SBH as compared to all other methods. The box sample sizes in the highest-risk box/group (out of $n = 250$ samples) were as follows for each method (and that replicate): SBH: $n_{SBH} = 39$, RST: $n_{RST} = 105$, RSF: $n_{RSF} = 124$, CPHR: $n_{CPHR} = 124$, SSPCA: $n_{SSPCA} = 124$, SSC: $n_{SSC} = 170$.

Overall, results from Figures 10 and 11 point out that the highest-risk box/group found by SBH is, as expected, smaller in size (support) and more extreme in terms of survival hazards (LHR) or risks, with consistent smaller event-free end-point times ($MEFT$) and probabilities ($MEFP$) than any other method/model under study. This was the goal. However, we did not expect the separation of the estimated survival distributions to be necessarily larger. In fact, the distributions of the log-rank test statistics (LRT) are not significantly different between most methods (except SSC). Interestingly, the Concordance Error Rates (CER) are slightly higher for SBH than most methods (except SSC). So, it's possible that the task of finding extreme survival/risks subgroups comes with some trade-off between achieving high levels of extremeness and high accuracy of survival time prediction.

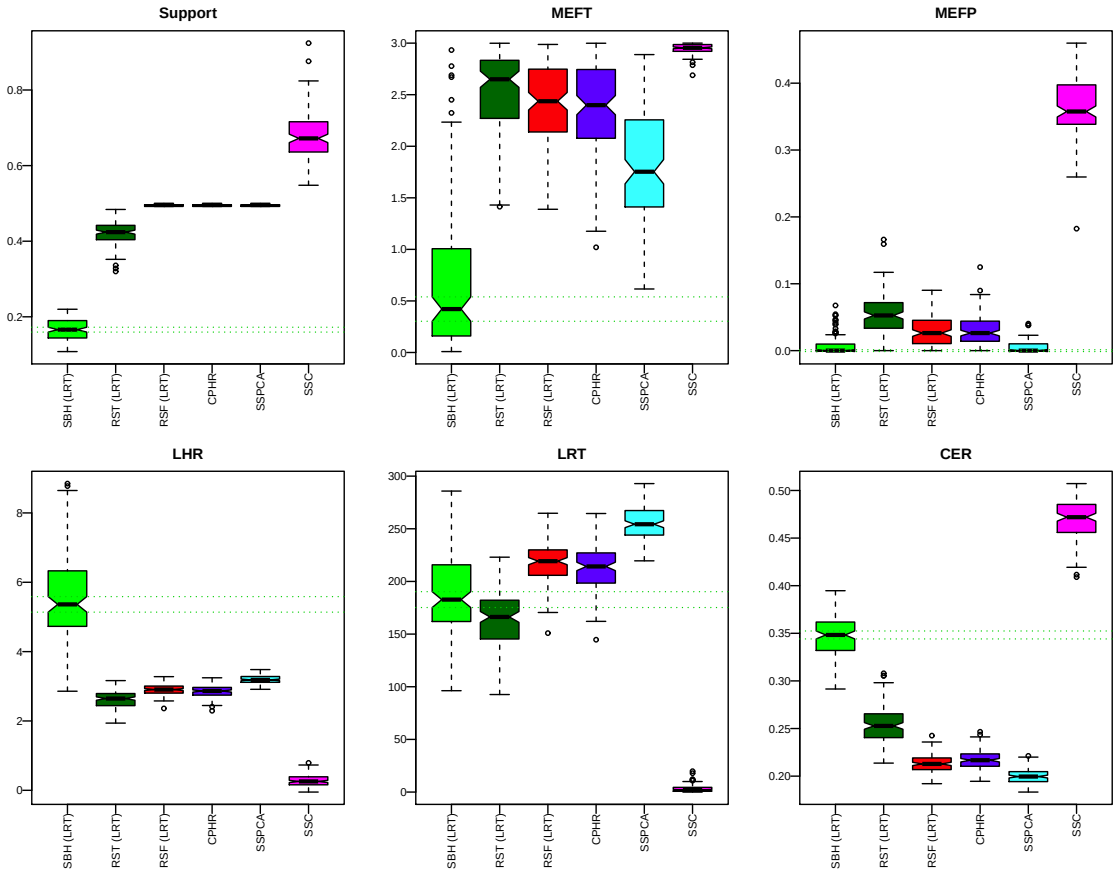


Figure 10: *Distributions of RCCV estimates of highest-risk/group end-points, computed over $B = 128$ repeated Monte-Carlo simulated models #1 and for all competitive non-parametric survival models under study. Comparisons include (i) Survival Bump Hunting (SBH), (ii) Regression Survival Trees (RST), (iii) Random Survival Forest (RSF), (iv) Cox Proportional Hazard Regression (CPHR), (v) Survival Supervised PCA (SSPCA), (vi) Survival Supervised Clustering (SSC). In parenthesis is shown the criterion used for peeling or partitioning as it applies. For each SBH boxplot, the pair of horizontal dotted lines delineates the approximate (95%) confidence interval of the median. Results are for the "Replicated Combined CV" (RCCV) technique and the LRT statistic used in the optimization criteria.*

4.4.3 Comparative Prediction Performance

The simulated survival model we drew from was according to model #1b as follows: samples contained within a box-shaped region R of the input covariate space had increased risk/hazards while samples outside of it had a uniform background risk. The survival times were generated as in section (4.1), using the exponential distribution with random uniform censoring. The regression function for samples within R was as in model #1.

Due to the rule-induction nature of our "Patient Recursive Survival Peeling" method (Algorithm 1), the box decision rule can be used as the classification rule. The cross-validated classification error is estimated from the discrepancies between the true and predictive classifications of the independent observations. Specifically, for each loop $k \in \{1, \ldots, K\}$ of the cross-validation, we compute a cross-validated estimate of the error by matching the SBH test-set "in-box" prediction samples to the true ones in the high-risk box-shaped region R of simulated model #1b using the left-out test-set $\mathcal{L}_k$. The final cross-validated estimate of the Miss-classification Error Rate (MER) is given by the average of the cross-validated errors from the K models $\{\tilde{\mathcal{R}}_k\}_{k=1}^{k=K}$ generated from each loop of the cross-validation. This is repeated B times to get variability estimates. Note that CER and MER, although related, are distinct: the MER evaluates the accuracy to predict that a sample will fall into (or outside) the highest-risk box/group found by a method/survival model, whereas the CER evaluates the accuracy to predict a sample survival time.

Prediction performances are then assessed using the usual accuracy metrics and Receiver Operating

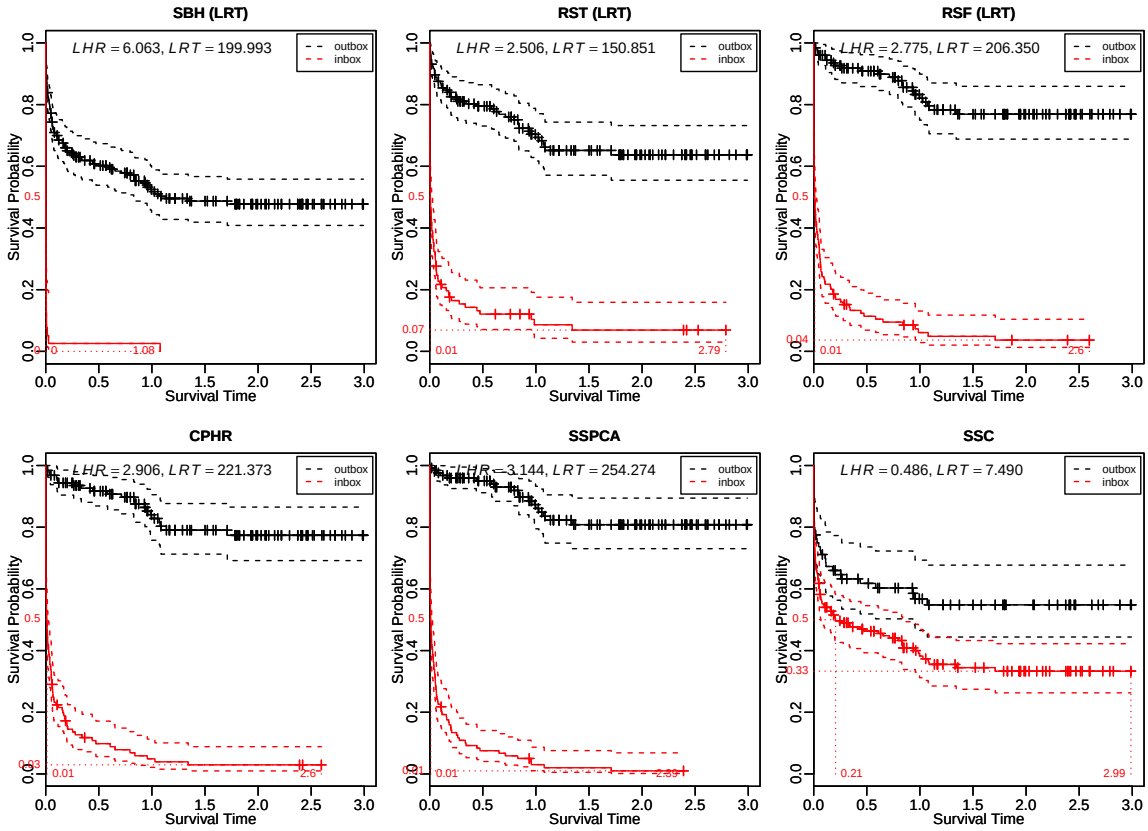


Figure 11: *Kaplan–Meier plots of RCCV survival probability curves for all competitive non-parametric survival models under study. Plots are illustrative of one replication out of $B = 128$. Comparisons include (i) Survival Bump Hunting (SBH), (ii) Regression Survival Trees (RST), (iii) Random Survival Forest (RSF), (iv) Cox Proportional Hazard Regression (CPHR), (v) Survival Supervised PCA (SSPCA), (vi) Survival Supervised Clustering (SSC). In parenthesis is shown the criterion used for peeling or partitioning as it applies. The "in-box" legends (red) corresponds to the highest-risk box/group. Cross-validated LRT, LHR of "in-box" samples are shown at the top of the plot for each method (and that replicate). Results are for the "Replicated Combined CV" (RCCV) technique and LRT statistic used in the optimization criteria.*

Characteristics (ROC) for each method and compared between them. For binary classes, a common metric for assessing the prediction performance is prediction accuracy through the use of True- and False-Positive Rates TPR and FPR, respectively, also known as $Sensitivity$ and $1 - Specificity$. By definition, the True- and False-Positive Rates are defined as:

$$TPR = Sensitivity = \frac{TP}{TP + FN}$$
$$FPR = 1 - Specificity = \frac{FP}{FP + TN}$$

where TP, FP, TN, FN stands for True-Positive, False-Positive, True-Negative and False-Negative, respectively. The performance of classification is naturally assessed by measuring the accuracy of prediction, whereas the performance of ranking is commonly measured by taking (AUC), the Area Under the Receiver Operating Characteristics (ROC) Curve (TPR versus FPR) [44]. An $AUC = 1$ corresponds to a perfect classifier, while an $AUC = 0.5$ corresponds to all possible performances of a random classifier. Finally, we also report Pearson's χ^2 contingency table test p-values (after continuity correction) of independence between the observed versus predicted counts. Table 4 reports the classification performance results of various survival models/methods in terms of contingency table test, area under the ROC curve and sensitivity/specificity.

Table 4: *Empirical χ^2 contingency table test p-values ($\widehat{P.val}$), Area Under the Curve ($\widehat{AUC}$), Sensitivity ($1-\widehat{FPR}$) and Specificity ($\widehat{TPR}$) for each method. Comparisons include (i) Survival Bump Hunting (SBH), (ii) Regression Survival Trees (RST), (iii) Random Survival Forest (RSF), (iv) Cox Proportional Hazard Regression (CPHR), (v) Survival Supervised PCA (SSPCA), (vi) Survival Supervised Clustering (SSC). Values are median estimates with standard errors of the sample mean in parenthesis. In parenthesis, next to the method, is also shown the criterion used for peeling or partitioning as it applies. Results are for the "Replicated Combined CV" (RCCV) technique and the LRT statistic used in the optimization criteria.*

Method	SBH (LRT)	RST (LRT)	RSF (LRT)	CPHR	SSPCA	SSC
$\widehat{P.val}$	0.000 (0.32)	0.909 (0.42)	0.065 (0.24)	0.037 (0.29)	0.037 (0.34)	0.519 (0.33)
$1-\widehat{FPR}$ $(Specificity)$	0.800 (0.11)	0.947 (0.04)	0.516 (0.01)	0.516 (0.01)	0.516 (0.01)	0.373 (0.05)
$\widehat{TPR}$ $(Sensitivity)$	1.000 (0.38)	0.125 (0.17)	1.000 (0.14)	1.000 (0.26)	1.000 (0.25)	0.833 (0.21)
$\widehat{AUC}$	0.899 (0.24)	0.533 (0.07)	0.757 (0.07)	0.758 (0.14)	0.758 (0.13)	0.591 (0.11)

Table 4 shows that SBH has a better prediction performance on all metrics than any other method/model under study. This directly results from its better trade-off of $Specificity$ and $Sensitivity$. Overall, this reflects the above results on comparative end-point estimates: SBH reaches out a more specific and smaller group of samples that is more extreme in survival hazards or risks. Conversely, other methods tend to be more sensitive, but way too un-specific. Note that this applies to Regression Survival Trees (RST) as well.

5 Real Data Analysis

Finally, we applied our Survival Bump Hunting (SBH) procedure to a publicly available real clinical dataset from the Women's Interagency HIV cohort Study (WIHS) [3]. It involves competing risks "AIDS/Death (before HAART)" and "Treatment Initiation (HAART)" during HIV-1 Infection in women. Here, for simplification purposes, only the first of the two competing events (the time to AIDS/Death) was used in our analysis. The data consists of $n = 485$ complete observations on the following $p = 4$ covariates in addition to the censoring indicator and (censored) time-to-event variables (Table 5).

Table 5: *Women's Interagency HIV Study (WIHS). Clinical dataset used with covariates description.*

Covariate Description	Range
AIDS/Death Diagnosis Time	T (years)
Event indicator variable	$C \in \{\text{AIDS/Dead} = 1, \text{Censored} = 0\}$
Patient age at time of FDA approval of first protease inhibitor	Age (years)
Injection Drug Users (IDU) history	$IDU \in \{\text{No history} = 0, \text{History} = 1\}$
Patients race	$Race \in \{\text{Other} = 0, \text{African-American} = 1\}$.
CD4 count	$CD4 \in [0, +\infty]/100$ cells/μl

All results in the WIHS clinical dataset were achieved using the "Replicated Combined CV" (RCCV) technique and the LRT statistic as peeling and optimization criterion, using $K = 5$, $A = 1024$ and $B = 128$. We show in Figure 12 the cross-validated tuning profile of LRT as a function of the number of peeling steps. According to our optimization criterion, we determined that the resulting "Replicated Combined CV" optimal length of the peeling trajectory is $\bar{L}^{rcv} = 5$ (Figure 12).

We show in Table 6 the cross-validated decision rules and highest-risk box/group statistics at each step. Note that the box sample size in the final highest-risk box/group is $n(l = 5) = 262$ out of a total sample size of $n = 485$. Here, the cross-validated trace of covariate usage is: $CD4$, Age, Age, Age (Table 6).

Finally, we show in Figure 13 the cross-validated Kaplan–Meir survival probability curves of the highest-risk group vs. lower-risk group at each step with their corresponding permutation p-values of separation. Notice how the curve separation increases with the peeling steps. The permutation p-values at each step are respectively: $\tilde{p}^{cv}(l = 0) = 1$, $\tilde{p}^{cv}(l = 1 - 5) \leqslant 9.7e - 5$ (Figure 13).

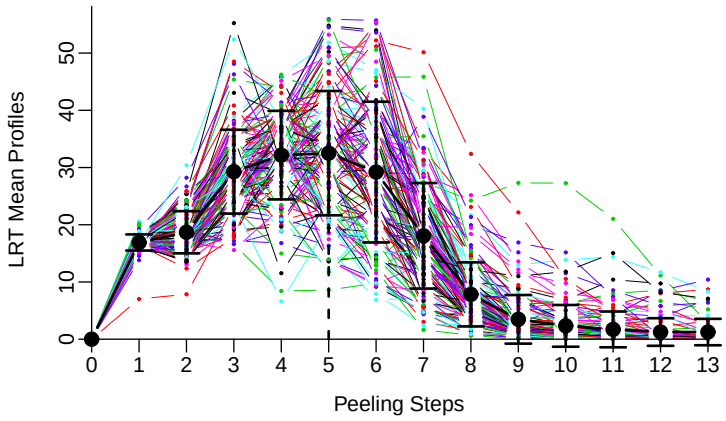


Figure 12: *Cross-validated tuning profile of the WIHS clinical dataset. The "Replicated Combined CV" cross-validated optimal peeling length ($\bar{L}^{rcv} = 5$) is shown with the vertical black dotted line. Each colored profile corresponds to one of the replications ($B = 128$). The cross-validated mean profile of the* LRT *statistic is shown by the solid black line with standard error of the sample mean.*

Table 6: *Cross-validated decision rules (top) and highest-risk box/group statistics (bottom) of the WIHS clinical dataset. Values are sample mean estimates with corresponding standard errors in parenthesis. The box sample size at each step is also shown. Step #0 corresponds to the situation where the starting box covers the entire test-set data $\mathcal{L}_k$ before peeling.*

Step l	*Age*	*IDU*	*Race*	*CD4*
0	$Age \geqslant 19.00$ (0.00)	$IDU \geqslant 0.00$ (0.00)	$Race \leqslant 1.00$ (0.00)	$CD4 \leqslant 19.33$ (0.00)
1	$Age \geqslant 19.00$ (0.00)	$IDU \geqslant 0.00$ (0.00)	$Race \leqslant 1.00$ (0.00)	$CD4 \leqslant 8.64$ (0.35)
2	$Age \geqslant 22.07$ (1.94)	$IDU \geqslant 0.00$ (0.00)	$Race \leqslant 1.00$ (0.00)	$CD4 \leqslant 8.51$ (0.35)
3	$Age \geqslant 23.30$ (2.65)	$IDU \geqslant 0.00$ (0.00)	$Race \leqslant 1.00$ (0.00)	$CD4 \leqslant 8.46$ (0.16)
4	$Age \geqslant 28.79$ (0.51)	$IDU \geqslant 0.00$ (0.00)	$Race \leqslant 1.00$ (0.00)	$CD4 \leqslant 7.66$ (0.83)
5	$Age \geqslant 29.22$ (0.53)	$IDU \geqslant 0.00$ (0.00)	$Race \leqslant 1.00$ (0.00)	$CD4 \leqslant 6.79$ (0.77)

Step l	$n(l)$	$\bar{\beta}^{rcv}(l)$	$\overline{T}_0'^{rcv}(l)$	$\overline{P}_0'^{rcv}(l)$	$\bar{\lambda}^{rcv}(l)$	$\bar{\chi}^{rcv}(l)$	$\bar{\theta}^{rcv}(l)$
0	485 (0.00)	1.00 (0.00)	10.8 (0.00)	0.17 (0.00)	0.00 (0.00)	0.00 (0.00)	1.00 (0.00)
1	436 (0.00)	0.90 (0.00)	10.8 (0.00)	0.17 (0.00)	0.61 (0.02)	16.90 (1.41)	0.46 (0.00)
2	398 (4.85)	0.82 (0.01)	10.8 (0.00)	0.16 (0.01)	0.54 (0.05)	18.69 (3.68)	0.45 (0.01)
3	364 (9.70)	0.75 (0.02)	10.8 (0.00)	0.14 (0.01)	0.61 (0.07)	29.28 (7.32)	0.43 (0.01)
4	325 (19.40)	0.67 (0.04)	10.8 (0.00)	0.13 (0.01)	0.61 (0.08)	32.17 (7.74)	0.42 (0.01)
5	262 (33.95)	0.54 (0.07)	10.8 (0.00)	0.11 (0.01)	0.61 (0.10)	32.51 (10.86)	0.42 (0.01)

The question of this study was whether it is possible to achieve a stratification or prognostication of patients for AIDS and HAART by using e.g. the 'Injection Drug Users' (IDU) history. Overall, SBH shows that it is possible to achieve a stratification and prognostication of patients that are more likely to be diagnosed or die of AIDS than others. The decision rule identifies a subgroup of $n(l = 5) = 262$ such patients that should be treated more aggressively than others, e.g. by putting them on HAART treatment sooner. In addition, SBH reveals that these patients are characterized by a lower $CD4$ count of $CD4 \lessapprox 6.79(\pm 0.77)/100$ cells/μl with an $Age \gtrapprox 29.22(\pm 0.53)$. Moreover, SBH reveals that 'Injection Drug Users' history (IDU) was actually *not* the most useful covariate at this stage of determination.

6 Discussion and Conclusion

To build a survival bump hunting model, fit by a recursive peeling procedure, we used two sets of criteria with different purposes: (i) the peeling criteria for model fitting by maximization of the rate of increase in *LHR*, *LRT* or *CHS* statistics (section 2.2.3), and (ii) the optimization criterion used for model tuning by maximization of the *LHR* or *LRT* or by minimization of the *CER* (section 3.2.3).

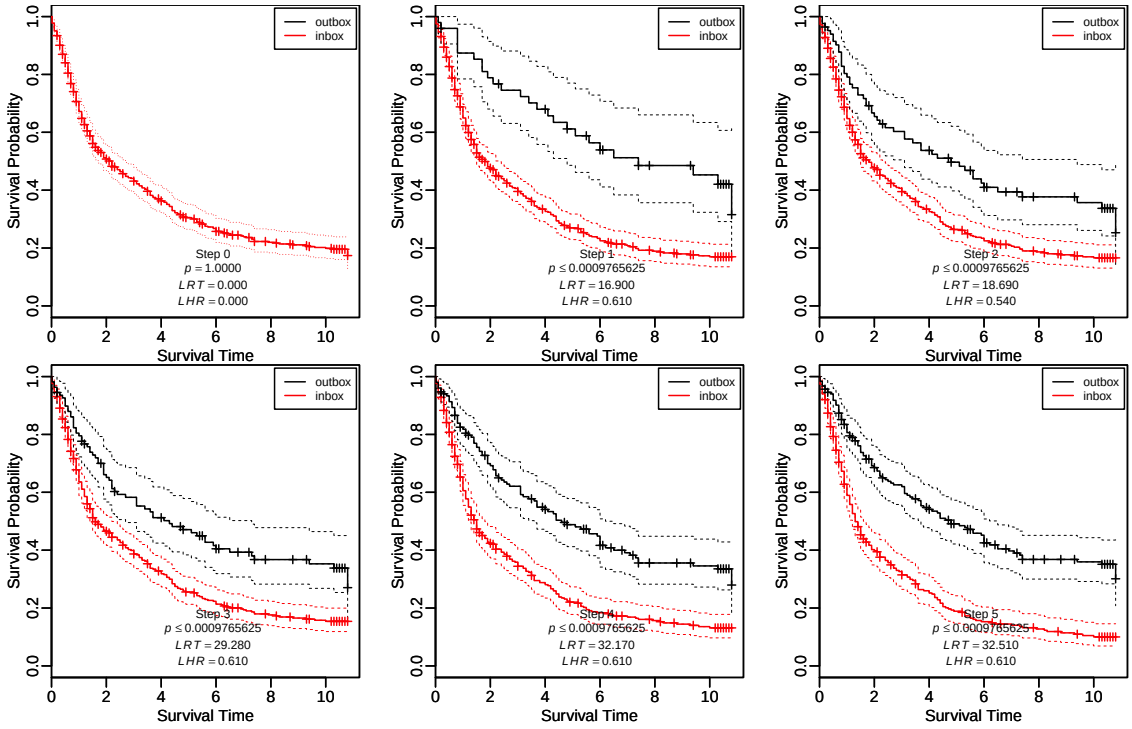


Figure 13: *Kaplan–Meier plots of RCCV survival probability curves of the WIHS clinical dataset. Each plot represents a step of the peeling sequence. Step #0 corresponds to the situation where the starting box covers the entire test-set data $\mathcal{L}_k$ before peeling. The "in-box" legends (red) corresponds to the highest-risk box/group. Cross-validated LRT, LHR and permutation p-values of "in-box" samples are shown at the bottom of the plot with the corresponding peeling step for each method. P-values $\hat{p}^{cv}(l) \leqslant 9.7e-5$ correspond to 1/10th of the precision limit (see section 3.5). Notice the single survival curve at Step #0 before peeling and how the survival curves of "in-box" and "out-of-box" samples separates as the peeling progresses.*

Despite being a well established concept as a splitting criterion in regression survival trees (section 2.2.3), one criticism about the *LRT* is that it tends to favor continuous variables and causes some uneven splits (end-cut preference). In this study, the main difference that we observed between the three peeling criteria used so far (section 4.3.2) is in the conservativeness of the number of peeling steps or peeling sequence length (model complexity), and to a lower extent, of the number of used covariates (model size), regardless of the dimensionality of the data. In summary, we recommended using as peeling criterion primarily the Cumulative Hazard Summary *CHS* and secondarily the Log-Rank Test *LRT* to induce more conservative estimates and reduce the risk of over-fitting, especially in high-dimensional data.

In contrast, we noted that the overall effect of the peeling criteria is relatively marginal compared to that of the optimization criterion. In this study, we showed that the choice of the optimization criteria is crucial and dependent on the dimensionality of the data. In summary, we recommended using the Concordance Error Rate *CER* as optimization criterion in every situation (low- and high-dimensional data), or alternatively the Log-Rank Test *LRT* in low-dimensional situation only (section 4.3.1).

Overall, both of our replicated cross-validation techniques, namely the "Replicated Combined CV" (RCCV) and "Replicated Averaged CV" (RACV) techniques, were found effective at controlling (at least in part) the overfitting and under-fitting issues, confirming that these techniques are appropriate to the task of survival bump hunting modeling by a recursive peeling procedure. However, we observed differences in the cross-validation techniques, which raised the questions whether RACV could lend to over-fitting more than RCCV, especially in high-dimensional data, and wether RACV performance could degrade faster than RCCV in situations with small sample sizes (see section 3.2.2). For these reasons, we recommended using RCCV preferably to RACV.

It is known that the stepwise covariate selection/usage procedure in the peeling loop of Algorithm 1

induces an inflation of variance estimates primarily because of the adaptive nature of the algorithm (each peeling step is conditional on the previous step). Using replicated cross-validation is therefore recommended to reduce the variability of cross-validation estimates in recursive peeling methods.

Survival estimates and model performance accuracy can be improved by the resampling technique used and resulting bias-variance trade-off. For instance, using a "larger" number of folds in K-fold CV in the presence of "small" number of events or samples is known to reduce bias, but also increase variance of estimates. Beside our cross-validation techniques, the Leave-one-Out Cross-Validation (LOOCV - Jackknife) or bootstrap-based resampling techniques are available, such as the ordinary bootstrap [26], the 0.632 / Out-of-Bootstrap Cross-Validation (0.632 OOBCV - [24]) or its latest variant (0.632+ OOBCV [27]). As per Efron, "*the ordinary bootstrap gives an estimate of error with low variability, but with a possibly large downward bias, particularly in highly over-fitted situations*" [24]. Conversely, cross-validation estimators are nearly unbiased (slightly upward), but have generally unacceptable high variability. For this reason, we chose to use a cross-validation procedure that could be replicated. The LOOCV and 0.632 OOBCV could be used instead of K-fold CV, but have larger variance estimators. An alternative may be the 0.632+ OOBCV estimator, which combines lower variance with a correction for bias [27].

Collectively, our peeling strategy, combined with our model tuning/selection method along with cross-validation techniques support the claim that optimal survival bump hunting modeling can be done. Further, results also indicate that our strategies help our "Patient Recursive Survival Peeling" (PRSP) algorithm (1) use the most informative covariates in the decision rule. This suggests the possibility of carrying out a joint internal variable selection with our PRSP procedure.

Some interesting differences between decision-tree and decision-box models lie in the weaknesses and strengths of the estimated solutions and in their applications. Here are some: (i) *Stratification*: if multiple groups are of interest, an advantage of recursive partitioning is that they directly lead to multi-group stratifications of the data, instead of just presenting a rule for a single high (low) vs. low (high) response group; (ii) *Interpretability*: binary decision trees lead to an intuitive hierarchical interpretation of groups that facilitates their interpretation unlike peeling methods that are not constrained to a tree structure; (iii) *Patience vs. Greediness*: recursive partitioning methods are however notoriously "greedy" (exponential decrease of the data as the space undergo partitioning based on typically binary split), but recursive peeling methods can be made "patient" at will (quantile-controlled decrease of the data), eventually helping recursive top-down peeling algorithms such as ours to better learn from the data.

7 Acknowledgments

This research was made possible with the contribution of the National Institute of Health (NIH), National Cancer Institute (NCI). J-E. Dazard and J. Sunil Rao were supported in part by NIH grant NCI R01-CA160593A1. Additional support came from NIH grant NCI P30-CA043703 of the Comprehensive Cancer Center at Case Western Reserve University (J-E. D). This work made use of the High Performance Computing Cluster in the Core Facility for Advanced Research Computing at Case Western Reserve University.

8 Supporting Information

- Supporting_Text; Supporting_Tables 1, 2; Supporting_Figures 1, 2, 3, 4, 5, 6.
- Supporting_Code: R package `PRIMsrc`, available at:
  CRAN repository: `https://cran.r-project.org/web/packages/PRIMsrc/index.html`
  GitHub repository: `https://github.com/jedazard/PRIMsrc`
- Supporting_Datasets: R package `PRIMsrc`, available at:
  CRAN repository: `https://cran.r-project.org/web/packages/PRIMsrc/index.html`
  GitHub repository: `https://github.com/jedazard/PRIMsrc/tree/master/data`

9 References

[1] Ahn, H. and Loh, W. Y. (1994), "Tree-structured proportional hazards regression modeling," *Biometrics*, 50, 471–85.

[2] Ambroise, C. and McLachlan, G. J. (2002), "Selection bias in gene extraction on the basis of microarray gene-expression data," *Proc Natl Acad Sci U S A*, 99, 6562–6.

[3] Bacon, M., von Wyl, V., and Alden, C. e. a. (2005), "Semi-Supervised Methods to Predict Patient Survival from Gene Expression Data," *Clin Diagn Lab Immunol*, 12, 1013–1019.

[4] Bair, E., Hastie, T., Paul, D., and Tibshirani, R. (2006), "Prediction by supervised principal components," *J Amer Stat Assoc*, 101, 119–137.

[5] Bair, E. and Tibshirani, R. (2004), "Semi-Supervised Methods to Predict Patient Survival from Gene Expression Data," *PLoS Biol.*, 2, 511–522.

[6] Baker, S., Kramer, B., and Srivastava, S. (2002), "Markers for early detection of cancer: Statistical guidelines for nested case-control studies," *BMC Medical Research Methodology*, 2, 4.

[7] Bohning, D. and Seidel, W. (2003), "Editorial: recent developments in mixture models," *Comp. Stat. Data Anal.*, 41, 349–357.

[8] Breiman, L. (2001), "Random forests," *Mach. Learn.*, 45, 5–32.

[9] Breiman, L., Friedman, J., Olshen, R., and Stone, C. (1984), *Classification and Regression Trees*, The Wadsworth statistics/probability series, Boca Raton, FLorida: Chapman and Hall/CRC.

[10] Burman, P. and Polonik, W. (2009), "Multivariate Mode Hunting: Data Analytic Tools with Measures of Significance," *Journal of Multivariate Analysis*, 100, 1198–1218.

[11] Ciampi, A., J., T., Nakache, J. P., and B., A. (1986), "Stratification by stepwise regression, correspondence analysis and recursive partition," *Comp. Stat. Data Anal.*, 4, 185–204.

[12] Cox, D. (1972), "Regression models and life-tables," *J. R. Stat. Soc.*, 30, 248–275.

[13] Davis, R. B. and Anderson, J. R. (1989), "Exponential survival trees," *Stat Med*, 8, 947–61.

[14] Dazard, J.-E., Choe, M., LeBlanc, M., and Rao, J. (2014), "Cross-Validated Survival Bump Hunting using Recursive Peeling Methods," in *JSM Proceedings. Section for survival methods for risk estimation/prediction*, Boston, MA, USA.: American Statistical Association, vol. IMS JSM, pp. 3366–3380.

[15] — (2016), "PRIMsrc for Identification and Characterization of Informative Prognostic Subgroups by Survival Bump Hunting," (submitted).

[16] Dazard, J.-E., Choe, M., LeBlanc, M., and Santana, A. (2015), "Contributed R Package PRIMsrc: PRIM Survival Regression Classification," The Comprehensive R Archive Network, https://cran.r-project.org/web/packages/PRIMsrc/index.html.

[17] Dazard, J.-E. and Rao, J. (2010), "Local Sparse Bump Hunting," *J. Comp. Graph. Statist.*, 19, 900–929.

[18] Dazard, J.-E., Rao, J., and Markowitz, S. (2012), "Local Sparse Bump Hunting Reveals Molecular Heterogeneity Of Colon Tumors," *Statistics in Medicine*, 31, 1203–1220.

[19] Diaz-Pachon, D., Dazard, J.-E., and Rao, J. (2015), "Unsupervised bump hunting using principal components," *(submitted)*, –.

[20] Diaz-Pachon, D., Rao, J., and Dazard, J.-E. (2015), "On the explanatory power of principal components," *(submitted)*, –.

[21] Dobbin, K., Beer, D., Meyerson, M., Yeatman, T., Gerald, W., Jacobson, J., Conley, B., Buetow, K., Heiskanen, M., Simon, R., Minna, J., Girard, L., Misek, D., Taylor, J., Hanash, S., Naoki, K., Hayes, D., Ladd-Acosta, C., Enkemann, S., Viale, A., and Giordano, T. (2005), "Interlaboratory comparability study of cancer gene expression analysis using oligonucleotide microarrays," *Clin Cancer Res*, 11, 565–572.

[22] Dobbin, K. and Simon, R. (2007), "Sample size planning for developing classifiers using high dimensional DNA microarray data," *Biostatistics*, 8, 101–117.

[23] Dupuy, A. and Simon, R. (2007), "Critical Review of Published Microarray Studies for Cancer Outcome and Guidelines on Statistical Analysis and Reporting," *J. Nat. Cancer Institute*, 99, 147–157.

[24] Efron, B. (1983), "Estimating the error rate of a predication rule: Improvement on cross-validation," *J Amer Stat Assoc*, 78, 316–331.

[25] Efron, B., Hastie, T., Johnstone, I., and Tibshirani, R. (2004), "Least angle regression," *The Annals of Statistics*, 32, 407–499.

[26] Efron, B. and Tibshirani, R. (1993), *An Introduction to the Bootstrap*, London: Chapman & Hall / CRC.

[27] — (1997), "Improvements on Cross-Validation: The .632+ Bootstrap Method," *J Amer Stat Assoc*, 92, 548–560.

[28] Ein-Dor, L., Kela, I., Getz, G., Givol, D., and Domany, E. (2005), "Outcome signature geness in breast cancer: is there a unique set?" *Bioinformatics*, 21, 171178.

[29] Fan, J., Han, F., and Liu, H. (2014), "Challenges of big data analysis," *National Science Review*, 1, 293–314.

[30] Fan, J. and Lv, J. (2008), "Sure independence screening for ultrahigh dimensional feature space," *J R Statist Soc B*, 70, 849–911.

[31] Friedman, J. and Fisher, N. (1999), "Bump hunting in high-dimensional data," *Statistics and Computing*, 9, 123–143.

[32] Gordon, L. and Olshen, R. A. (1985), "Tree-structured survival analysis," *Cancer Treat Rep*, 69, 1065–9.

[33] Haibe-Kains, B., El-Hachem, N., Birkbak, N., Jin, A., Beck, A., Aerts, H., and Quackenbush, J. (2013), "Inconsistency in large pharmacogenomic studies," *Nature*, 504, 389–394.

[34] Harrell, F. E., J., Califf, R. M., Pryor, D. B., Lee, K. L., and Rosati, R. A. (1982), "Evaluating the yield of medical tests," *JAMA : the journal of the American Medical Association*, 247, 2543–6.

[35] Hartigan, J. and Mohanty, S. (1992), "The RUNT Test for Multimodality," *Joumal of Classification*, 9, 63–70.

[36] Hastie, T., Tibshirani, R., and Friedman, J. (2009), *The Elements of Statistical Learning: Data Mining, Inference, and Prediction*, New York: Springer Science.

[37] Hothorn, T. and Lausen, B. (2003), "On the exact distribution of maximally selected rank statistics," *Comput. Statist. Data Analysis*, 43, 121137.

[38] Huang, J., Ma, S., and Xie, H. (2006), "Regularized estimation in the accelerated failure time model with high dimensional covariates," *Biometrics*, 62, 813820.

[39] Il-Gyo, C. and Chi-Hyuck, J. (2008), "Flexible patient rule induction method for optimizing process variables in discrete type," *Expert Syst. Appl.*, 34, 3014–3020.

[40] Ishwaran, H., Kogalur, U., Blackstone, E., and Lauer, M. (2008), "Random survival forests," *The Annals of Applied Statistics*, 2, 841–860.

[41] Ishwaran, H. and Rao, J. S. (2005), "Spike and slab variable selection: frequentist and Bayesian strategies," *Ann Statist*, 33, 730–773.

[42] Kalbfleisch, J. and Prentice, R. (2002), *The Statistical Analysis of Failure Time Data.*, Wiley Series in Probability and Statistics, Hoboken, NJ, USA: Wiley, 2nd ed.

[43] Kehl, V. and Ulm, K. (2006), "Responderidentification in clinical trials with censored data," *Comput. Statist. Data Anal.*, 50, 1338–1355.

[44] Klement, W. and Flach, P. (2008), "Soft Receiver Operating Characteristics Curves," in *Proceedings of the 3rd International Workshop on Evaluation Methods for Machine Learning*, ICML.

[45] LeBlanc, M. and Crowley, J. (1992), "Relative risk trees for censored survival data," *Biometrics*, 48, 411–25.

[46] — (1993), "Survival trees by goodness of split," *J Amer Stat Assoc*, 88, 457–67.

[47] LeBlanc, M., Jacobson, J., and Crowley, J. (2002), "Partitioning and peeling for constructing prognostic groups," *Stat Methods Med Res*, 11, 247–74.

[48] LeBlanc, M., Moon, J., and Crowley, J. (2005), "Adaptive Risk Group Refinement," *Biometrics*, 61, 370–378.

[49] Leek, J., Scharpf, R., Bravo, H., Simcha, D., Langmead, B., Johnson, W., Geman, D., Baggerly, K., and Irizarry, R. (2010), "Tackling the widespread and critical impact of batch effects in high-throughput data," *Nat Rev Genet*, 11, 733–739.

[50] Liu, X., Minin, V., Huang, Y., Seligson, D., and Horvath, S. (2004), "Statistical Methods for Analyzing Tissue Microarray Data," *J. Pharm. Stat.*, 14, 671–685.

[51] Markatou, M., H., T., S., B., and G., H. (2005), "Analysis of Variance of Cross-Validation Estimators of the Generalization Error," *J. Machine Learning Research*, 6, 1127–1168.

[52] McShane, L., Cavenagh, M., Lively, T., Eberhard, D., Bigbee, W., Williams, P., Mesirov, J., Polley, M.-Y., Kim, K., Tricoli, J., Taylor, J., Shuman, D., Simon, R., Doroshow, J., and Conley, B. (2013), "Criteria for the use of omics-based predictors in clinical trials: explanation and elaboration," *BMC Medicine*, 11, 220.

[53] McShane, M., Cavenagh, M., Lively, T., Eberhard, D., Bigbee, W., Mickey Williams, P., Mesirov, J., Polley, M.-Y., Kim, K., Tricoli, J., Taylor, J., Shuman, D., Simon, R., Doroshow, J., and Conley, B. (2013), "Criteria for the use of omics-based predictors in clinical trials," *Nature*, 502, 317–320.

[54] Michiels, S., Koscielny, S., and Hill, C. (2005), "Prediction of cancer outcome with microarrays: a multiple random validation strategy," *Lancet*, 365, 488–492.

[55] Molinaro, A., Simon, R., and Pfeiffer, R. (2005), "Prediction error estimation: a comparison of resampling methods," *Bioinformatics*, 21, 3301–3307.

[56] Naftel, D., E., B., and M., T. (1985), "Conservation of events," Research report.

[57] Ntzani, E. and Ioannidis, J. (2003), "Predictive ability of {DNA} microarrays for cancer outcomes and correlates: an empirical assessment," *The Lancet*, 362, 1439 – 1444.

[58] Polonik, W. (1995), "Measuring Mass Concentration and Estimating Density Contour Clusters: an Excess Mass Approach," *The Annals of Statistics*, 23, 855–881.

[59] Polonik, W. and Wang, Z. (2010), "PRIM Analysis," *Journal of Multivariate Analysis*, 101, 525–540.

[60] Ransohoff, D. (2004), "Rules of evidence for cancer molecular marker discovery and validation," *Nature Reviews Cancer*, 4, 309–314.

[61] Rozal, G. and Hartigan, J. (1994), "The MAP Test for Multimodality," *Journal of Classification*, 11, 5–36.

[62] Segal, M. R. (1988), "Regression Trees for Censored Data," *Biometrics*, 44, 35–47.

[63] Shi, L., Reid, L., Jones, W., Shippy, R., Warrington, J., Baker, S., Collins, P., de Longueville, F., Kawasaki, E., Lee, K., Luo, Y., Sun, Y., Willey, J., Setterquist, R., Fischer, G., Tong, W., Dragan, Y., Dix, D., Frueh, F., Goodsaid, F., Herman, D., Jensen, R., Johnson, C., Lobenhofer, E., Puri, R., Schrf, U., Thierry-Mieg, J., Wang, C., Wilson, M., and Consortium, M. (2006), "The MicroArray Quality Control (MAQC) project shows inter- and intraplatform reproducibility of gene expression measurements," *Nat Biotechnol*, 24, 1151–1161.

[64] Simon, R., Radmacher, M., Dobbin, K., and McShane, L. (2003), "Pitfalls in the Use of DNA Microarray Data for Diagnostic and Prognostic Classification," *J. Nat. Cancer Institute*, 95, 14–18.

[65] Simon, R., Subramanian, J., Li, M.-C., and Menezes, S. (2011), "Using cross-validation to evaluate predictive accuracy of survival risk classifiers based on high-dimensional data," *Briefings in Bioinformatics*, 12, 203–214.

[66] Subramanian, J. and Simon, R. (2010), "Gene expression-based prognostic signatures in lung cancer: ready for clinical use?" *J. Natl. Cancer Inst.*, 102, 464474.

[67] — (2011), "An evaluation of resampling methods for assessment of survival risk prediction in high-dimensional settings," *Stat. Med.*, 30, 642653.

[68] — (2013), "Overfitting in prediction models Is it a problem only in high dimensions?" *Contemporary Clinical Trials*, 36, 636641.

[69] Therneau, T., Grambsch, P., and Fleming, T. (1990), "Martingale based residuals for survival models," *Biometrika*, 77, 147–160.

[70] Tibshirani, R. (1996), "Regression shrinkage and selection via the Lasso," *J R Statist Soc*, 58 (Series B), 267–288.

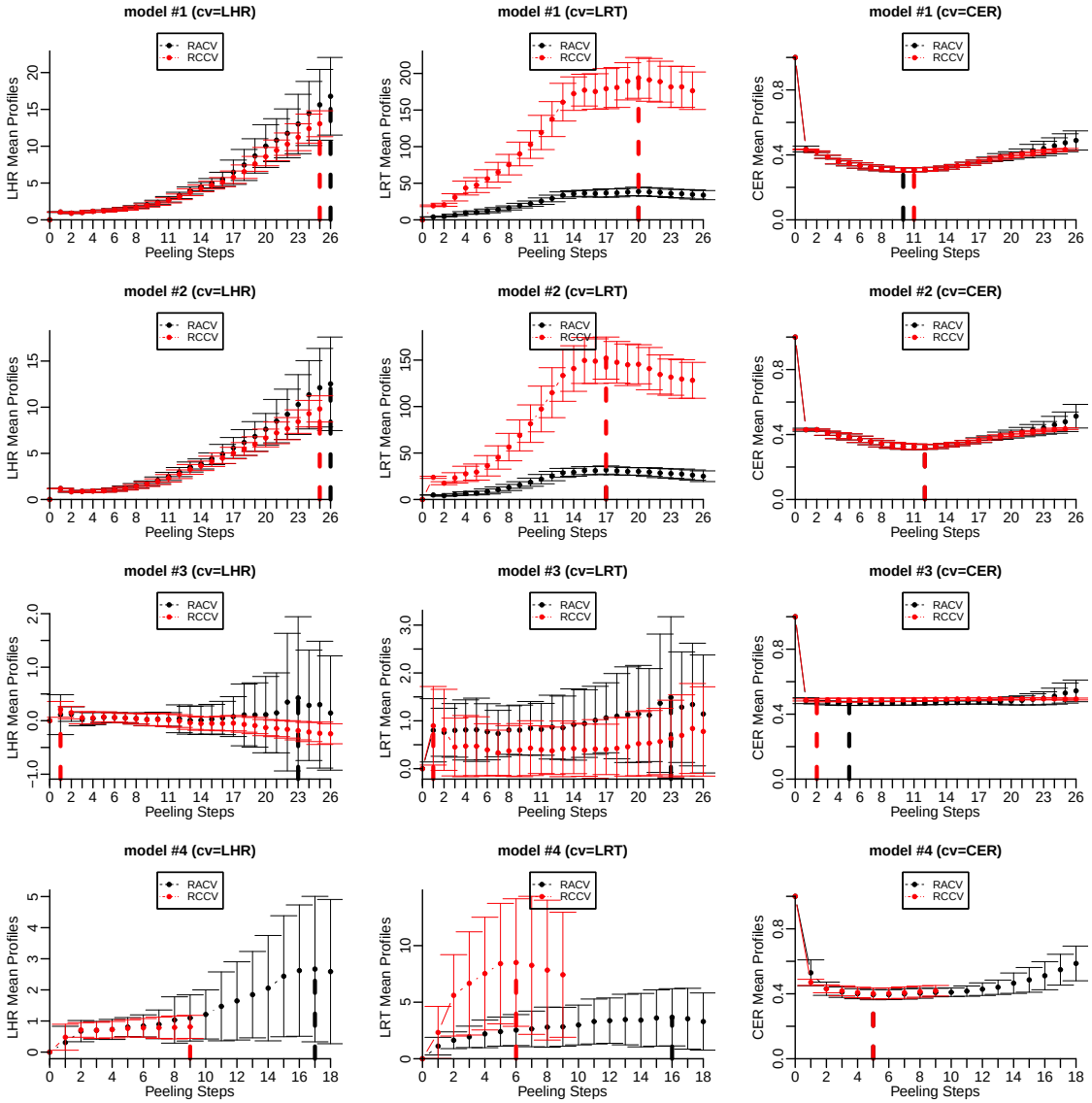
[71] Varma, S. and Simon, R. (2006), "Bias in error estimation when using cross-validation for model selection," *BMC bioinformatics*, 7, 91–99.

[72] Wang, P., Kim, Y., Pollack, J., and Tibshirani, R. (2004), "Boosted PRIM with Application to Searching for Oncogenic Pathway of Lung Cancer," in *Computational Systems Bioinformatics Conference, International IEEE Computer Society*, IEEE Computer Society, pp. 604–609.

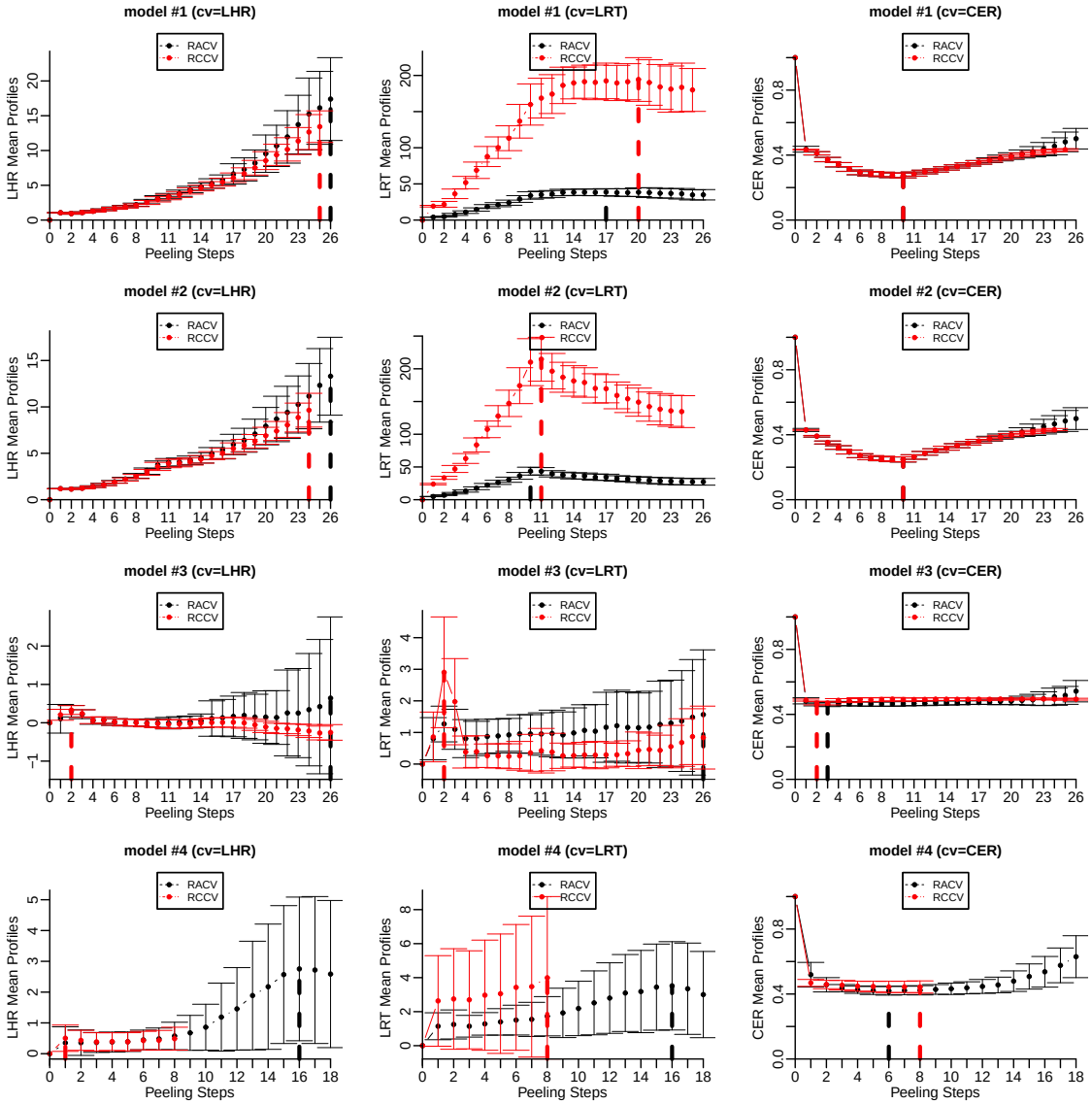
[73] Wu, L. and Chipman, H. (2003), "Bayesian Model-Assisted PRIM Algorithm," Tech. rep., Departments of Statistics and Actuarial Science, University of Waterloo.

[74] Zou, H. and Hastie, T. (2005), "Regularization and variable selection via the elastic net," *J R Statist Soc*, 67 (Series B), 301–320.

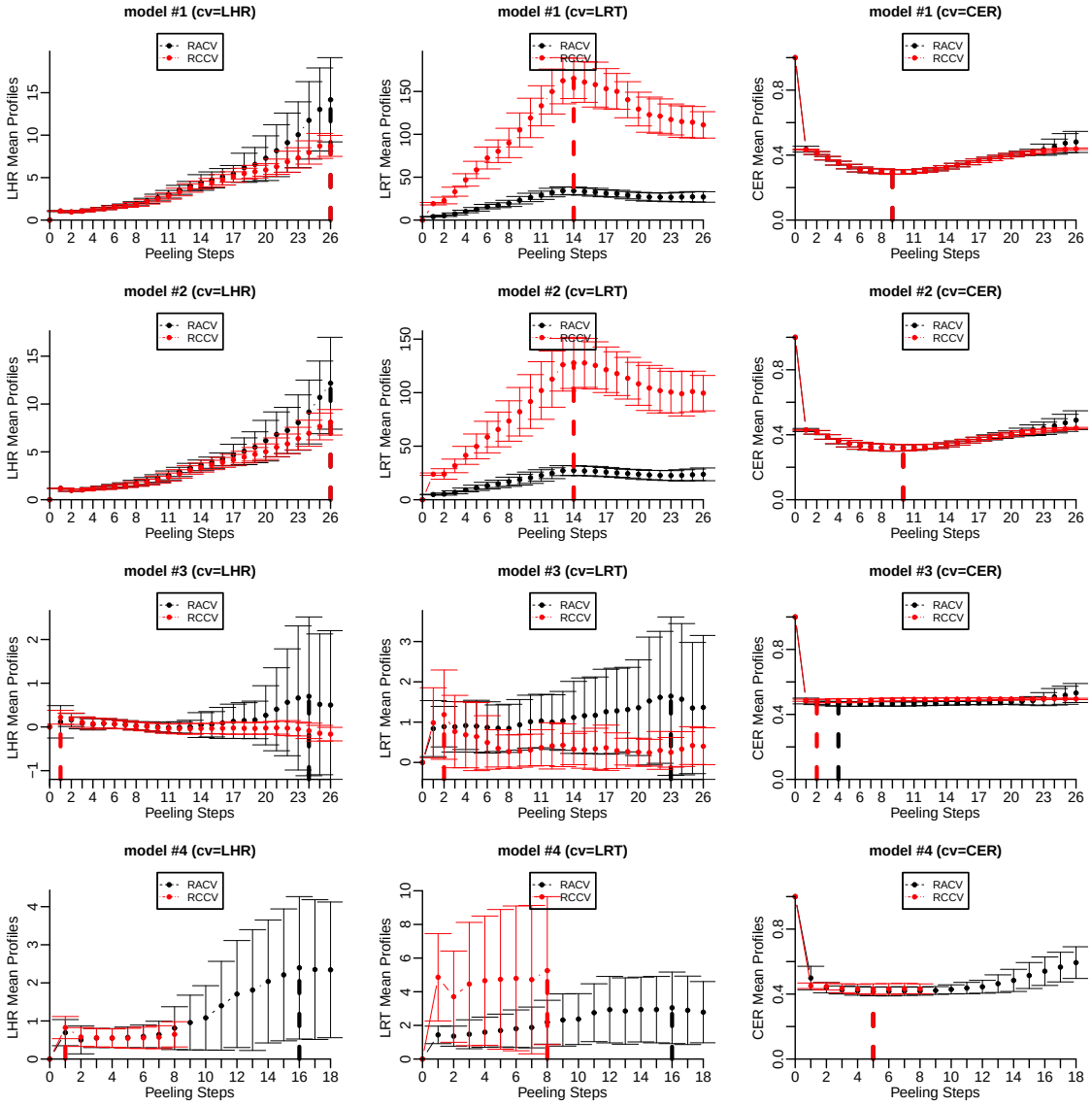
SUPPORTING INFORMATION



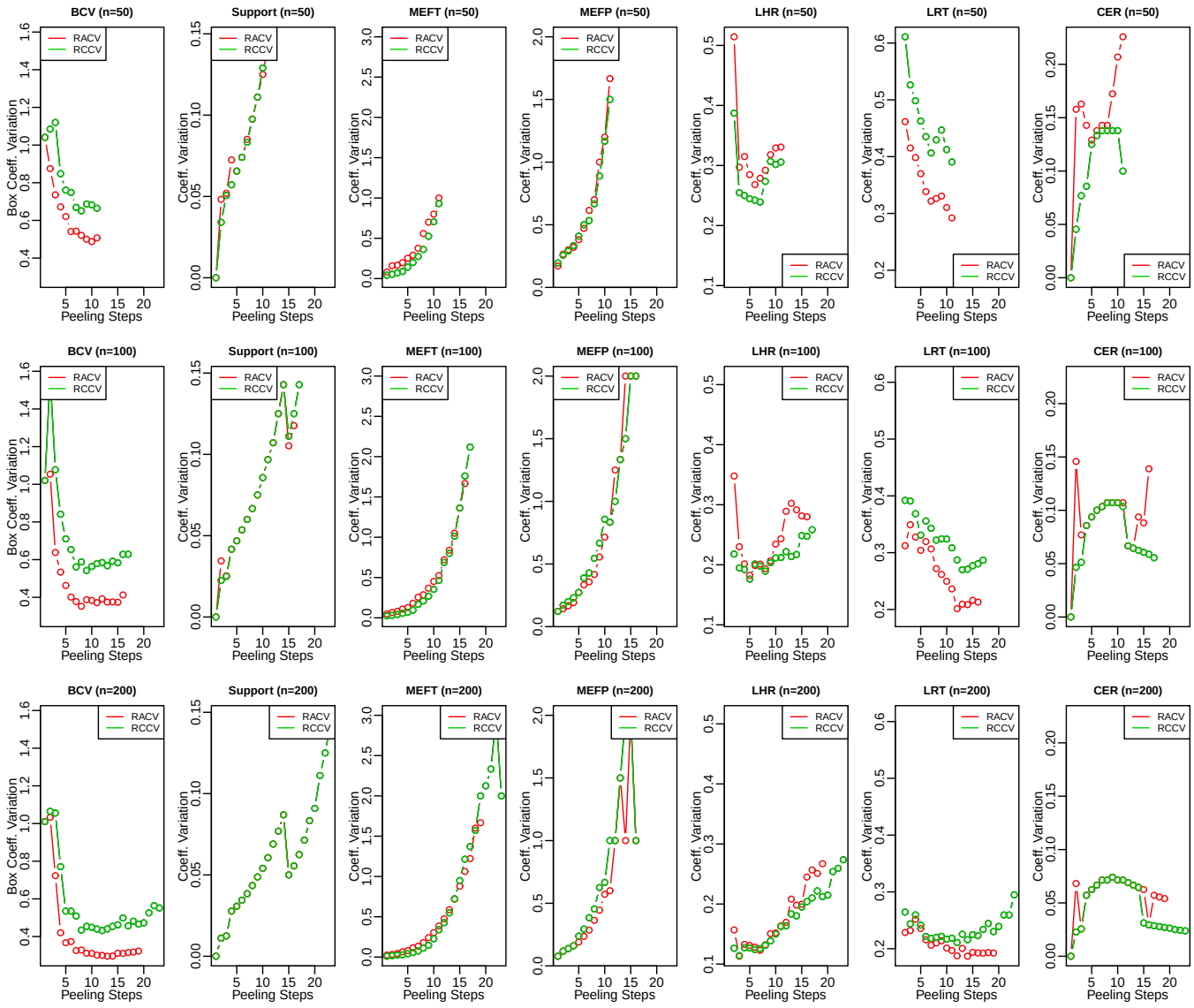
Supporting_Figure 1: *Comparison of cross-validated tuning profiles of box end-point statistics between cross-validation techniques (overlaid: "Replicated Averaged CV" or RACV (black) vs. "Replicated Combined CV" or RCCV (red)) and optimization criteria (by rows: Log Hazard Ratio (LHR), Log-Rank Test (LRT) or Concordance Error Rate CER) in a given simulation model (by columns: simulation models #1, #2, #3 or #4). Results are for the Log Hazard Ratio (LHR) peeling criterion. The resulting "Replicated CV" optimal peeling length* $\bar{L}^{rcv}$ *(see eq. 19) of the peeling trajectory is shown in each case (vertical dashed lines). Each dotted line corresponds to a cross-validated mean profile of the statistic used in the optimization criterion with the corresponding standard error of the sample mean, both calculated over the replications* ($B = 128$)*. Notice the situations of cross-validation success or failure as described in section 4.3.1 and Figure 4. Also, notice the expected increase of variance of cross-validated point estimates towards the right-end of the profiles corresponding to an increase in model uncertainty and regions of risk of overfitting.*



Supporting_Figure 2: *Comparison of cross-validated tuning profiles of box end-point statistics between cross-validation techniques (overlaid: "Replicated Averaged CV" or RACV (black) vs. "Replicated Combined CV" or RCCV (red)) and optimization criteria (by rows: Log Hazard Ratio (LHR), Log-Rank Test (LRT) or Concordance Error Rate CER) in a given simulation model (by columns: simulation models #1, #2, #3 or #4). Results are for the Log-Rank Test (LRT) peeling criterion. The resulting "Replicated CV" optimal peeling length* $\bar{L}^{rcv}$ *(see eq. 19) of the peeling trajectory is shown in each case (vertical dashed lines). Each dotted line corresponds to a cross-validated mean profile of the statistic used in the optimization criterion with the corresponding standard error of the sample mean, both calculated over the replications* ($B = 128$)*. Notice the situations of cross-validation success or failure as described in section 4.3.1 and Figure 4. Also, notice the expected increase of variance of cross-validated point estimates towards the right-end of the profiles corresponding to an increase in model uncertainty and regions of risk of overfitting.*



Supporting_Figure 3: *Comparison of cross-validated tuning profiles of box end-point statistics between cross-validation techniques (overlaid: "Replicated Averaged CV" or RACV (black) vs. "Replicated Combined CV" or RCCV (red)) and optimization criteria (by rows: Log Hazard Ratio (LHR), Log-Rank Test (LRT) or Concordance Error Rate CER) in a given simulation model (by columns: simulation models #1, #2, #3 or #4). Results are for the Cumulative Hazard Summary (CHS) peeling criterion. The resulting "Replicated CV" optimal peeling length* $\bar{L}^{rcv}$ *(see eq. 19) of the peeling trajectory is shown in each case (vertical dashed lines). Each dotted line corresponds to a cross-validated mean profile of the statistic used in the optimization criterion with the corresponding standard error of the sample mean, both calculated over the replications* ($B = 128$)*. Notice the situations of cross-validation success or failure as described in section 4.3.1 and Figure 4. Also, notice the expected increase of variance of cross-validated point estimates towards the right-end of the profiles corresponding to an increase in model uncertainty and regions of risk of overfitting.*



Supporting Figure 4: *Profiles of coefficient of variation of Box Coefficient of Variation (BCV), survival end-points and prediction performance metrics. Comparative coefficient of variation profiles are shown for situations with decreasing sample sizes* $n \in \{50, 100, 200\}$. *Results are for simulated model #1 and the LRT statistic used in both peeling and optimization criteria.*

Supporting_Table 1: *Effect of peeling and optimization criteria as well as cross-validation techniques on the cross-validated numbers of used covariates by the PRSP algorithm (see Algorithm 1) out of the total number of pre-selected ones (brackets). Numbers are reported for the combined effects of: (i) peeling criteria (by rows: Log Hazard Ratio (LHR), Log-Rank Test (LRT) or Cumulative Hazard Summary (CHS)), (ii) optimization criteria (by columns: Log Hazard Ratio (LHR), Log-Rank Test (LRT) or Concordance Error Rate (CER)), (iii) cross-validation techniques (by columns: "Replicated Averaged CV" or RACV and "Replicated Combined CV" or RCCV), and (iv) the four tested simulation models (by rows: Model #1, #2, #3 or #4).*

Model #1

		Optimization Criterion					
		LHR		LRT		CER	
		RACV	RCCV	RACV	RCCV	RACV	RCCV
Peeling Criterion	LHR	3[3]	3[3]	3[3]	3[3]	3[3]	3[3]
	LRT	3[3]	3[3]	3[3]	3[3]	2[3]	2[3]
	CHS	3[3]	3[3]	3[3]	3[3]	3[3]	3[3]

Model #2

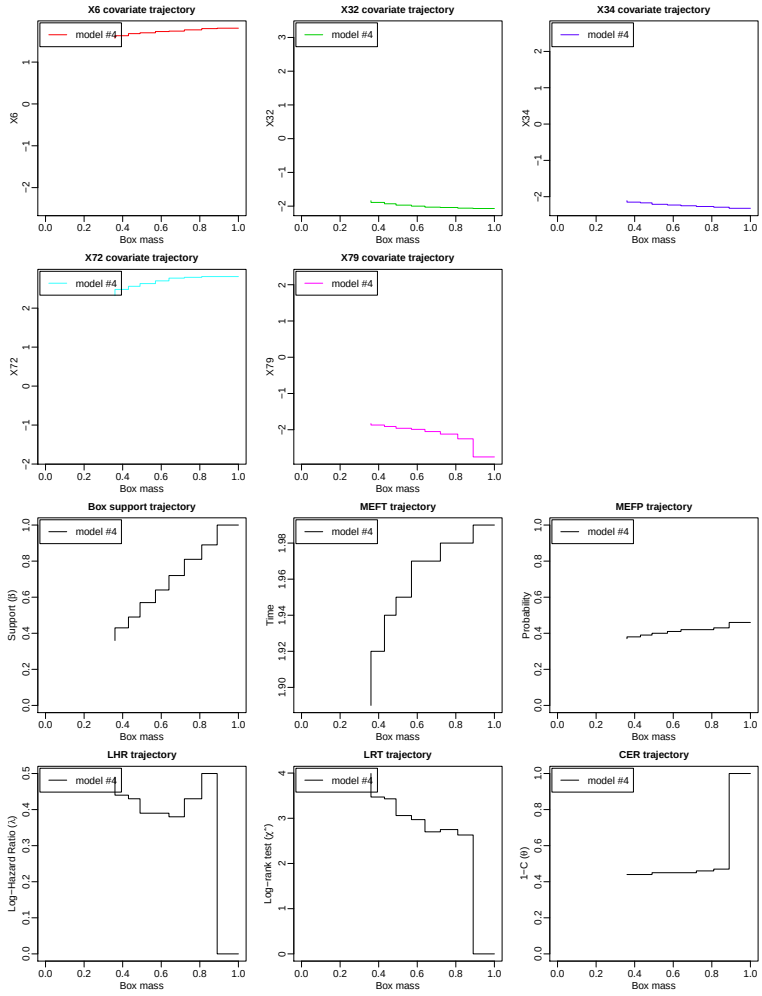
		Optimization Criterion					
		LHR		LRT		CER	
		RACV	RCCV	RACV	RCCV	RACV	RCCV
Peeling Criterion	LHR	3[3]	3[3]	3[3]	3[3]	3[3]	3[3]
	LRT	3[3]	3[3]	2[3]	2[3]	2[3]	2[3]
	CHS	3[3]	3[3]	3[3]	3[3]	3[3]	3[3]

Model #3

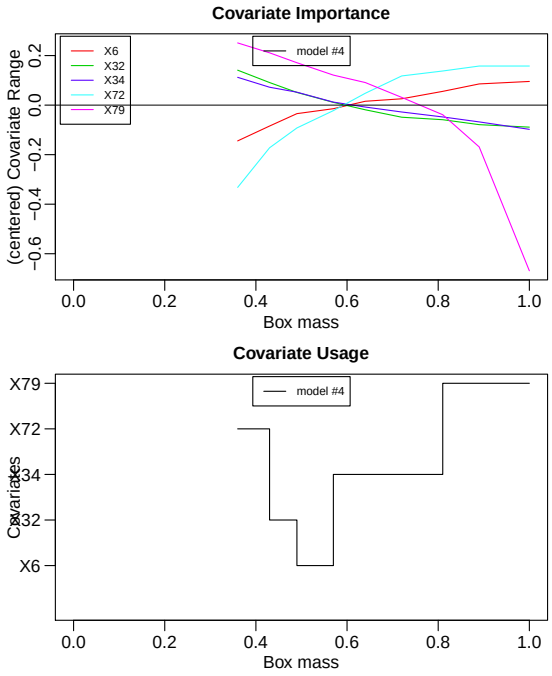
		Optimization Criterion					
		LHR		LRT		CER	
		RACV	RCCV	RACV	RCCV	RACV	RCCV
Peeling Criterion	LHR	3[3]	1[3]	3[3]	1[3]	3[3]	2[3]
	LRT	3[3]	2[3]	3[3]	2[3]	3[3]	2[3]
	CHS	2[3]	1[3]	2[3]	2[3]	2[3]	2[3]

Model #4

		Optimization Criterion					
		LHR		LRT		CER	
		RACV	RCCV	RACV	RCCV	RACV	RCCV
Peeling Criterion	LHR	11[352]	6[352]	10[352]	3[352]	3[352]	3[352]
	LRT	8[352]	1[352]	8[352]	5[352]	3[352]	5[352]
	CHS	3[352]	1[352]	3[352]	3[352]	3[352]	3[352]



Supporting_Figure 5: *Comparison of replicated combined cross-validated results for the peeling trajectories in simulated model #4 for the "Replicated Combined CV" (RCCV) technique and the Cumulative Hazard Summary CHS as peeling criterion and the Concordance Error Rate CER as optimization criteria. Notice that covariates* $(\mathbf{x}_6, \mathbf{x}_{32}, \mathbf{x}_{34}, \mathbf{x}_{72}, \mathbf{x}_{79})$ *were those effectively used by the PRSP algorithm (see Algorithm 1) out of* $p = 1000$ *total covariates and* $p = 100$ *informative ones (see simulation design 4.1).*



Supporting_Figure 6: *Comparison of replicated combined cross-validated trace plots of covariate importance* $\overline{VI}(l)$ *(top) and covariate usage* $\overline{VU}(l)$ *(bottom) in simulated model #4 for the "Replicated Combined CV" (RCCV) technique and the Cumulative Hazard Summary CHS as peeling criterion and the Concordance Error Rate CER as optimization criteria. Notice that covariates* $(\mathbf{x}_6, \mathbf{x}_{32}, \mathbf{x}_{34}, \mathbf{x}_{72}, \mathbf{x}_{79})$ *were those effectively used by the PRSP algorithm (see Algorithm 1) out of* $p = 1000$ *total covariates and* $p = 100$ *informative ones (see simulation design 4.1).*

Supporting_Table 2: *Comparison of cross-validated decision rules (upper Supporting Table) and box end points statistics of interest (lower Supporting Table) in simulated model #4 for the "Replicated Combined CV" (RCCV) technique and the Cumulative Hazard Summary CHS as peeling criterion and the Concordance Error Rate CER as optimization criteria. For conciseness, only the initial and final decision rules (*$\bar{L}^{rcv}$*th step) are shown. Step #0 corresponds to the situation where the starting box covers the entire test-set data* $\mathcal{L}_k$ *before peeling. Values are sample mean estimates with corresponding standard errors in parenthesis. Notice that covariates* $(\mathbf{x}_6, \mathbf{x}_{32}, \mathbf{x}_{34}, \mathbf{x}_{72}, \mathbf{x}_{79})$ *were those effectively used by the PRSP algorithm (see Algorithm 1) out of* $p = 1000$ *total covariates and* $p = 100$ *informative ones (see simulation design 4.1).*

	Step l	$\mathbf{x}_6$	$\mathbf{x}_{32}$	$\mathbf{x}_{34}$	$\mathbf{x}_{72}$	$\mathbf{x}_{79}$
model #4	0	$\mathbf{x}_6 \leqslant 1.81$ (0.00)	$\mathbf{x}_{32} \geqslant -2.07$ (0.00)	$\mathbf{x}_{34} \geqslant -2.32$ (0.00)	$\mathbf{x}_{72} \geqslant 2.81$ (0.00)	$\mathbf{x}_{79} \geqslant -2.75$ (0.00)
	1	$\mathbf{x}_6 \leqslant 1.80$ (0.43)	$\mathbf{x}_{32} \geqslant -2.06$ (0.03)	$\mathbf{x}_{34} \geqslant -2.29$ (0.06)	$\mathbf{x}_{72} \geqslant 2.81$ (0.04)	$\mathbf{x}_{79} \geqslant -2.25$ (0.30)
	⋮	⋮	⋮	⋮	⋮	⋮
	8	$\mathbf{x}_6 \leqslant 1.57$ (0.25)	$\mathbf{x}_{32} \geqslant -1.84$ (0.23)	$\mathbf{x}_{34} \geqslant -2.11$ (0.20)	$\mathbf{x}_{72} \geqslant 2.32$ (0.50)	$\mathbf{x}_{79} \geqslant -1.83$ (0.40)

	Step l	$n(l)$	$\bar{\beta}^{rcv}(l)$	$T_0'^{rcv}(l)$	$P_0'^{rcv}(l)$	$\bar{\lambda}^{rcv}(l)$	$\bar{\chi}^{rcv}(l)$	$\bar{\theta}^{rcv}(l)$
model #4	0	100 (0.00)	1.00 (0.00)	1.99 (0.00)	0.46 (0.00)	0.00 (0.00)	0.00 (0.00)	1.00 (0.00)
	1	89 (2.00)	0.89 (0.02)	1.98 (0.01)	0.43 (0.02)	0.50 (0.43)	2.63 (2.66)	0.47 (0.02)
	⋮	⋮	⋮	⋮	⋮	⋮	⋮	⋮
	8	36 (6.00)	0.36 (0.06)	1.89 (0.16)	0.37 (0.07)	0.49 (0.37)	3.99 (4.78)	0.44 (0.04)